FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

# Improving Time of Arrival Estimation Using Encoded Acoustic Signals

**João Miguel Fernandes Magalhães**


MASTER IN ELECTRICAL AND COMPUTERS ENGINEERING

Supervisor: Prof. Dr. José Carlos Santos Alves

Second Supervisor: Prof. Dr. Nuno Alexandre Lopes Moreira da Cruz

July 8, 2018

# Improving Time of Arrival Estimation Using Encoded Acoustic Signals

**João Miguel Fernandes Magalhães**

MASTER IN ELECTRICAL AND COMPUTERS ENGINEERING

July 8, 2018

# Resumo

Sistemas de localização subaquática são cruciais para explorar o mar. Os sistemas mais comuns para este ambiente usam sinais acústicos para comunicação e estimação de posição. Existem vários desafios envolvidos no desenvolvimento destes sistemas. Esta dissertação incidirá principalmente sobre a parte destes sistemas que estima o momento exato de chegada dum sinal acústico. Este aspeto é crucial para estabelecer uma posição relativa precisa.

Para melhorar a estimativa do momento de chegada de um sinal acústico propõe-se o uso de sequências binárias pseudo-aleatórias moduladas em BPSK. As sequências binárias pseudo-aleatórias usadas têm um único pico de autocorrelação e baixos resultados de correlação cruzada. Tendo em conta estas propriedades, espera-se que o pico de correlação indique um momento de chegada preciso.

É implementado também um sistema de comunicação de baixa taxa binaria. Isto é possível porque o sistema usa um conjunto de 16 sequências binárias pseudo-aleatórias com boas propriedades de correlação cruzada. Este conjunto de sequências binárias origina um conjunto de sinais com igualmente boas propriedades de correlação cruzada. Ao ter no recetor uma unidade de correlação para cada um dos 16 possíveis sinais, o sinal recebido é identificável e portanto, dados podem ser transferidos se cada sinal tiver um símbolo previamente atribuído.

Após o momento de chagada do sinal ser estimado, esse resultado é convertido em distância com base na velocidade do som na água. No tanque de testes do laboratório, onde o transmissor e o receptor estavam a menos de 2 m de distância, o pior desvio padrão obtido para uma posição foi 1.68 cm. Esse desvio padrão baixa para 0.34 mm se for reduzido o número de medições feitas por segundo. Em testes de campo, para uma distância de 118 m o desvio padrão da distância estimada foi 1.69 cm. Em todos os testes, nenhum erro ocorreu na identificação de qual sinal foi recebido, o que significa que não ocorreram erros na transmissão de dados, a máxima taxa de dados transferidos foi 32 bit/s.

# Abstract

Underwater localisation systems are crucial to explore the sea. The most common systems for this environment use acoustic signals for communication and position estimation. There are several challenges involved in developing these systems. This dissertation will primarily focus on the part of these systems that estimates the exact moment an acoustic signal has arrived. This is crucial to establish an accurate relative position.

To improve the time of arrival estimation of an acoustic signal it is proposed the use of pseudo-random binary sequences modulated in Binary Phase Shift Keying (BPSK). The used pseudo-random binary sequences have a single auto-correlation peak and low cross-correlation results. Given these properties, it is expected that the correlation peak renders a precise time of arrival.

It is also implemented a low bit rate communication system from the transmitter to the receiver. This is possible because the system uses a set of 16 pseudo-random binary sequences with good cross-correlation properties. This set of binary sequences originates a set of signals with equally good cross-correlation properties. By having one correlator for each of the possible signals at the receiver, the received signal is identifiable, and data is transferred by, in advance, attributing a symbol to each signal.

After the time of arrival is estimated, that result is translated to distance based on the underwater sound speed. In the laboratory test tank, where transmitter and receiver were less than 2 m apart, the worst standard deviation obtained for a position was 1.68 cm. This standard deviation can decrease to 0.34 mm by reducing the number of estimation results per second. In field tests, for a distance of 118 m the standard deviation of the estimated distance was 1.69 cm. In all testes, no errors occurred in the identification of which signal was received, and therefore no communication errors, the maximum tested bit rate was 32 bit/s.

# Agradecimentos

Em primeiro lugar, agradeço ao professor José Carlos Alves por todo o apoio, não só ao longo do desenvolvimento da dissertação mas sobre tudo pelo facto de se preocupar com experiencia de aprendizagem de todos os alunos que passam pelo nosso curso. Agradeço também professor Nuno Alexandre Lopes Moreira da Cruz por disponibilizar material

Quero também agradecer aos meus pais por todo o apoio que me deram ao longo dos 5 anos.

Finalmente, agradeço à FEUP por todos os amigos que fiz ao longo destes 5 anos.

João Magalhães

*"You should be glad that bridge fell down.*
*I was planning to build thirteen more to that same design"*

Isambard Kingdom Brunel

# Contents

# List of Figures

# List of Abbreviations

| | |
|---|---|
| BPSK | Binary Phase Shift Keying |
| LBL | Long Base Line |
| SBL | Short Base Line |
| USBL | Ultra Short Base Line |
| RSSI | Received Signal Strength Indicator |
| ToF | Time of Flight |
| ToA | Time of Arrival |
| INS | Inertial Navigation System |
| GPS | Global Positioning System |
| GIB | GPS Intelligent Buoy |
| VRU | Vertical Reference Unit |
| SNR | Signal to Noise Ration |
| TDoA | Time Difference of Arrival |
| DoA | Direction of Arrival |
| PN | Pseudo-Noise |
| ISI | Inter Symbol Interference |
| PRBS | Pseudo-Random Binary Sequence |
| ADC | Analog to Digital Converter |
| FPGA | Field Programmable Gate Array |
| SoC | Sytem on a Chip |
| QPSK | Quadrature Phase Shift Keying |
| PPS | Pulse Per Second |

# Chapter 1

# Introduction

> "The ocean is the lifeblood of Earth, covering more than 70% of the planet's surface, driving weather, regulating temperature, and ultimately supporting all living organisms. Throughout history, the ocean has been a vital source of sustenance, transport, commerce, growth, and inspiration." NOAA (2017)

## 1.1 Context

Even though the ocean has a significant influence in everything, from the air we breathe to the climate patterns, we know very little about it. Most of the knowledge we do have about the ocean is limited to areas with shallower waters. The deep areas of the ocean remain mostly unexplored, even though we rely more and more on them for food, energy sources and other resources

Ocean exploration is used to understand better the planet that we live in, to search for the unexpected and the unusual. However, exploring the ocean doesn't consist in merely wondering about in it in hopes of finding something new. "It is disciplined, systematic, and includes rigorous observations and documentation of biological, chemical, physical, geological, and archaeological aspects of the ocean." NOAA (2017)

Exploring the deepest part of the ocean provides high-value environmental intelligence needed to address both current and emerging science and management needs. Exploring helps to ensure that the ocean's resources are well managed so that the current generation can enjoy them as well as the future.

Some of the benefits of exploring the deep-sea include.

1. Better understand environmental changes;
2. Possible discovery of resources with use in medical drugs;
3. Discovery of food sources;
4. Discovery of energy sources;

"Yet even as the importance of deep areas of the ocean in our everyday lives continues to increase, our knowledge of these areas remains limited – in many instances, we are "flying blind" when it comes to management, regulation, and resource use in deep-water areas." NOAA (2014)

When on a mission to explore the deep-sea, an underwater vehicle needs to be able to place itself geographically, or at least to be tracked by the controller. Explored territory needs to be marked as such, the vehicle in mission needs to know the course it is taking, and new discoveries need to be positioned geographically. So it is obvious the importance of localisation systems in underwater environments.

Systems used on land, such as GPS, are highly accurate and reliable. However, those rely on electromagnetic signals which are evilly attenuated on water. A different approach is needed for this environment, the most used systems rely on acoustic signals. Acoustic signals propagate further than electromagnetic signals on water, but this approach is not without its challenges. Those challenges will be explored in the following chapters.

The work here presented has the goal of helping in the quest for the exploration of one of the most important and underexplored ecosystems on the planet, the ocean.

## 1.2 Objectives and Approach

This work intends to improve the underwater localisation systems presently used in the autonomous underwater vehicles (AUV) developed in INESC TEC, and also implement a one way low bit rate communication system. The system implemented in those AUV's works based on the time of arrival estimation of an acoustic signal.

With the help of a high precision and low drift clock reference, the underwater vehicle knows when a signal is transmitted, and by determining the signals time of arrival, it is possible to determine the distance of the receiver relative to the transmitter. Such a system will only ever be as precise as the precision it has when estimating the signal time of arrival.

The starting point to improve the estimation of time of arrival is to send pseudo-random binary sequences modulated in Binary Phase Shift Keying (BPSK). At the reception end, samples will be correlated with the expected acoustic signals. A correlation peak corresponds to the arrival instant.

The use of pseudo-random binary sequences is motivated by the possibility of working with a set of sequences that have high autocorrelation and low cross-correlation. This property facilitates the identification of the transmitted acoustic signal.

Because it is possible to develop the work in a field-programmable gate array (FPGA), there is a lot of computational power available. Several correlators can work in parallel, this allows the use of several pseudo-random sequences to create several different acoustic signals with high autocorrelation and low cross-correlation, making them easily identifiable by the receiver.

By using a set of binary sequences with the stated properties, it is possible to attribute a symbol to each one, and therefore, use them to communicate, (for example with 16 pseudo-random binary sequences is possible to create 16 distinct sound signals where each of the different signal represents a symbol of four bits).

# Chapter 2

# State of the Art

This chapter explores the parts involved in underwater navigation and communication. Presents the systems used to establish a geographical position underwater and the components used in those systems.

Because the point of this work is not to develop a new localisation system but to improve the already existing ones, the work then focuses on the more relevant matter, which is, the time of arrival estimation.

## 2.1 Underwater Vehicles

There are three main types of underwater vehicles.

1. **Manned Submersibles**;
2. **Remotely Operated Vehicles (ROVs)**;
3. **Autonomous Underwater Vehicles (AUVs)**;

Manned Submersibles were the first to appear, built in 1775 and first used in 1776. An appearance mainly motivated by military reasons.

The next step was to use these vehicles to explore the ocean, and that is when ROV's and AUV's were first developed and used.

ROV's and AUV's don't carry people, therefore, are more compact, easier to deploy, less expensive and can reach deeper parts of the ocean.

AUV's are capable of performing long duration and long-range missions with complete energy self-sufficiency. These vehicles can go a very long time without surfacing. Time used during descent and ascent is time wasted. This creates a case for the development of high accuracy, low power consumption and low-cost localisation systems.

## 2.2 Localisation and Navegation Systems

The following, are some characteristics that developers must take into account when designing and producing underwater localisation systems: Tan et al. (2011)

- **High Accuracy** — Accuracy is always important in localisation systems, exactly how precise the system needs to be depends on the requirements defined by the user;
- **Fast Convergence** — Given that underwater vehicles and their reference points are constantly drifting in the water, it is imperative that the localisation systems converge rapidly, otherwise, once they determine a position, that result is already outdated;
- **Low Communication Costs** — This point refers to energy consumption costs, depending on the vehicle in question, this can be battery powered and used in long missions, in which case, power consumption management is of high priority;
- **Good Scalability** — In the case of acoustic localisation systems, the high propagation delay and high power attenuation, make these systems highly dependable on the number of reference points used to establish an accurate relative position. Consequently, an underwater localisation protocol should be distributed and depend on as few as possible reference points;

The remainder of this section presents underwater localisation techniques other than based on acoustic signals.

Acoustic localisation architectures are presented in depth in 2.3 due to its higher relevance to the work here presented.

### 2.2.1 Inertial Navigation Systems

An Inertial Navigation System (INS), computes the position, velocity and attitude(roll, pitch and heading) using three accelerometers and three gyros. The three accelerometers are mounted perpendicularly to each other, each measuring acceleration relative to inertial space. The three gyros measure the angular rate relative to the inertial space. Rowan (2008)

So, by knowing the original absolute position at the beginning of the mission and by computing its movement along the way, it is possible for an underwater vehicle to know its absolute position at all times.

Position drift rates, for high-quality commercial grade INS units are on the order of few kilometres per day Defense (2018), but, the cost, power consumption, and size have historically made INS systems unattractive for small AUVs.

### 2.2.2 Geographical Positioning

These systems rely on a previously acquired data-set that characterises several aspects of a geographical location. The system reads the sight as best as it can and then compares what was

observed with the information present in the database. When a match is found, the system is able to establish its position.

- **Optical** — This category of geographical positioning system relies on camera images. Then it compares the sight images with does on its database.

  Limitations for optical systems in underwater environments include the reduced range of the cameras, susceptibility to scattering, and inadequacy of lighting. As a result, visible wavelength cameras are more commonly installed on hovering AUVs because they can get close to objects of interest. Also, visual odometry relies on the existence of features. Therefore, optical underwater navigation methods are particularly well suited to small-scale mapping of feature-rich environments. Examples include ship hulls or shipwreck inspections; Paull et al. (2014)

- **Sonar** — Sonar imaging of the ocean predates AUVs, this system works very much like the optical system, but instead of using optical images it uses acustic signals.

  The quality of system, much like in the case of the optical system, depends on the number and quality of features. These systems emit a fan-shaped been of acoustic signals and record the reflections in a series cross-track slices. When mosaicked together, these assembled slices form an image of the seabed within the swath of the beam. Hard objects protruding from the seabed send a strong return which is represented as a dark image. Shadows and soft areas, such as mud and sand, send weaker returns which are represented as lighter images. Paull et al. (2014)

The biggest drawback of these systems lays on the fact that a data-set needs to be created prior to its utilization. So, when it comes to exploring unknown areas, these systems don't translate into viable options.

## 2.3 Acoustic Localisation and Communication systems

Given that the focus of this work in acoustic underwater localisation systems, this section will concentrate exclusively on such systems.

### 2.3.1 Channel Modelling

Devising underwater localisation and communication systems presents a particularly difficult challenge due to the properties of the underwater acoustic channel. For that reason it is essential to have an accurate channel model. Unfortunately, good statistical models for simulation are currently lacking.

The characteristics of the channel are mainly influenced by the following factors: path loss, time-varying multipath, and low propagation speed. Another factor to take into account when designing these systems is the Doppler effect, caused by the relative movement between transmitter

and receiver. To go about developing accurate systems for underwater environments, these factors must be studied in depth, the remainder of this section explores each one individually.

The theory presented in this section is explained in Cabrala (2014) and supported by several papers there referred.

#### 2.3.1.1 Propagation speed

The estimation of the underwater sound speed is crucial to every existing localisation system devised for this environment.

The sound speed underwater is usually situated between 1440 and 1550 m/s. However, small variations on the conditions have a profound effect on acoustic propagation. The sound speed underwater is mainly affected by pressure, salinity and temperature, taking into account these parameters the sound speed can be estimated using the equation 2.1.

$$C = 1449.2 + 4.6T + (1.34 - 0.01T)(S - 35) + 0.016z \quad \textit{Khanet al.}\ (2014) \tag{2.1}$$

Where C is the sound speed in seawater, T is the temperature in Celsius, S is the salinity in parts per thousands and z is the depth of the water.

The propagation speed increases both with the increase of local pressure and density. Generally, in ocean water, for the first 200 meters as the depth increases the speed reduces due to the decrease of the temperature. Past the 200 meters, as the depth increases the temperature stabilizes, and with it, the sound speed steadily increases.

Figure 2.1 shows the variation of the sound speed as the depth changes. For more information on the subject refer to Khan et al. (2014) and Aparicio et al. (2015).

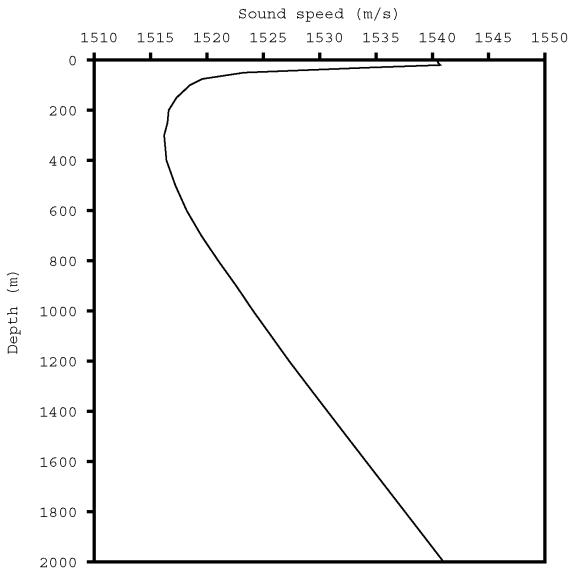



Figure 2.1: Sound speed profile underwater

#### 2.3.1.2 Attenuation and Noise

Underwater, the attenuation of acoustic signals is due to the transformation of these signals into heat. This affects the signals strength and results in path loss due to the degradation of the amplitude of the signal. Moreover, the signal also experiences spread loss which increases with distance.

The overall path loss is approximated by:

$$A(L,f) = L^k \alpha f^L \text{ Khanet al. (2014)} \tag{2.2}$$

Where L is the transmission distance, f is the signal frequency and k is the spreading loss. Alpha is the absorption coefficient in $dB/Km$ and its value can be obtained from Thorp's equation:

$$\alpha = 0.11\frac{f^2}{1+f^2} + 44\frac{f^2}{4100+f^2} + 2.75\times10^{-4}f^2 + 0.003 \text{ Khanet al. (2014)} \tag{2.3}$$

Where f is the signal frequency in KHz.

Besides attenuation, the noise also affects the ability of successfully transmitting sound signals due to its impact in SNR. The ambient noise in the acoustic channel is usually characterised as Gaussian, but not white since its power spectral density decays at around 18 dB per decade. Cabrala (2014)

#### 2.3.1.3 Time-varying multipath

Multipath occurs due to the variation of the sound speed underwater and also because of the geometric boundaries of the ocean, (seabed and surface).

"Multipath arises due to both refraction and reflection of sound. Reflections occur on the waveguide boundaries, and they are only specular under very particular conditions. Refraction happens on account of the varying speed of sound, and causes the arrival of a theoretically unlimited number of rays even from a point source. The combined arrivals from the two effects make the channel reverberant and dispersive, and lead to a finite multipath spread on the order of tens of milliseconds, the delay of the longest significant path." Cabrala (2014)

Figure 2.2 illustrates the problem, the sea surface and the seafloor create reflections that reach the receiver possibly in phase. Because these other signals travel across different paths, they arrive at the receiver at random moments. The reflections may even arrive before the direct signal, which hinders the detection of the direct signal.

Multipath raises problems mainly on communication systems, causing high Inter-Symbol Interference (ISI), because of the delayed signals. Multipath is more severe in shallower waters due to their proximity to the surface boundary. Less of a problem in deeper waters.

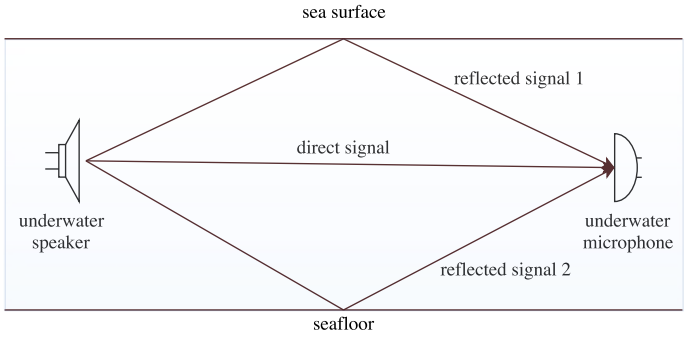



Figure 2.2: Signal path

#### 2.3.1.4 Doppler effect

Doppler effect results from relative movement between the transmitter and the receiver, which causes frequency shifts. This problem is aggravated by the low speed of the sound. The magnitude of this effect is determined by $\alpha = v/c$, where $c$ is the speed of the sound and $v$ is the relative speed of the parts involved. When using electromagnetic signals, because of their high propagation speed, this problem is only taken into account when in the presence of extremely high relative speed, rotation of the earth, jets, etc.

### 2.3.2 Acoustic Equipment

Acoustic localisation systems use several different electronic components, some used to emit signals, some to receive and some used for both, here are presented the most relevant ones and their use.

- **Transducer** — Emitter and receiver, transforms one form of energy into another, (electric to acoustic and vice versa);
- **Transponder** — Receiver and emitter. Sends a response to a received signal;
- **Pinger** — Emitter that periodically sends a predefined signal;
- **Hydrophone** — Receiver mounted on underwater vehicle or another reception structure, i.e. underwater microphone;
- **Responder** — Emitter whos signal will be received by hydrophone or transponder;

### 2.3.3 Signal Range Estimation

The following methods are used to estimate the distance to the signal source. Tan et al. (2011)

- **Received Signal Strength Indicator (RSSI)** — each node estimates its distance to the signal source by examining the signal strength and comparing it to a range dependent signal

model. However, this models can't be relied on for high accuracy localisation systems. As explained in 2.3.1.2, the path loss in underwater acoustic channels varies with several factors, and multipath can result in considerable energy fading. RSSI is not the primary choice for underwater localisation systems;

- **Time of Flight (ToF)** — this approach requires a transducer and a transponder, the transducer transmits a signal, upon reception, the transponder sends a response. Once that response reaches the transducer, the round-trip-time is obtained, and the distance is estimated using the estimation of the sound speed at the sight. As previously discussed, the sound speed on the water is susceptible to variations which can cause accuracy problems for this method;

- **Time of Arrival (ToA)** — this technique requires the receiver to know when the signal is transmitted. Upon receiving the acoustic signal, the receiver calculates the delay, and, once again, using the estimated underwater sound speed, calculates the distance to the reference point.

### 2.3.4 Localisation Systems Architecture

There are three broad classes of underwater acoustic positioning systems. These are, Long Base Line (LBL), Short Base Line (SBL) and finally Ultra Short Base Line (USBL). Infrastructures in mission establish their positions relative to a framework of baseline stations. This framework must be deployed before operations.

#### 2.3.4.1 Long Base Line (LBL)

The LBL architecture involves the use of a framework of transponder beacons displaced in the seabed before any mission. It also requires the infrastructure in the mission to have an attached acoustic transducer. The distances between beacons within the calibrated network are designated baselines, and range from 50 to 2000 meters. Rowan (2008)

The transducer, attached to the vehicle in mission, transmits an acoustic signal which is detected by the transponders situated on the seabed. The transponders transmit a response, and when the response reaches the vehicle in mission, the vehicle obtains the round trip time. From the round trip time, it is possible to calculate the distance between the transducer and each of the transponders.

The minimum number of transponders required for a unambiguous navigation is three. However, more than three provide a degree of redundancy and quality check.

In figure 2.3 is depicted an example of this system. The letter A represents the vehicle in mission, B, C, D and E are the transponder beacons attached to the seabed. As explained before, the vehicle can establish a relative distance to each one of the transponders. Once that process is complete, the vehicle can calculate its absolute position using trilateration.

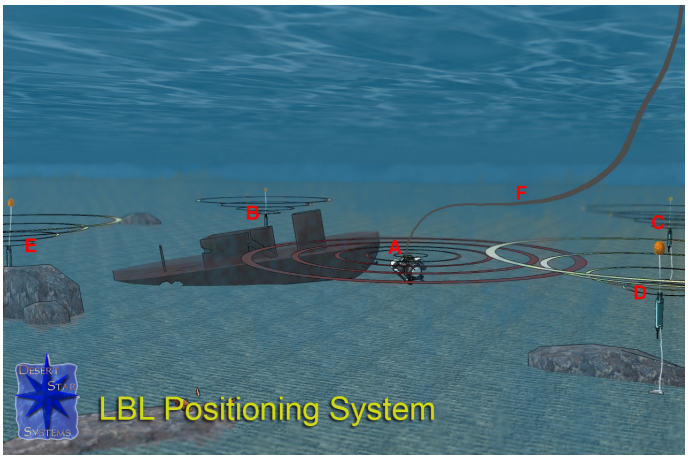



Figure 2.3: Long Base Line Architecture Example

For a higher level of accuracy, this system is sometimes paired with inertial navigation. INS systems were presented in 2.2.1.

This technique yields a very high accuracy, even over very long distances, generally better than 1 meter of precision, sometimes as good as 0.01 meters. Surveyor et al. (2013)

GPS Intelligent Buoy (GIB), is an approach derived from LBL that may be classified as inverted LBL. In this approach, the transducers are installed on GPS equipped sonobuoys that are either drifting in the sea surface ore moored. The sound source is tracked using the ToA method explained in 2.3.3. Typically, several GIBs are deployed over a given area of operation. The actual number varies depending on the size of the test area and the desired accuracy. This system is patent protected by the French company ACSA-underwater-GPS. For more information about this system refer to Surveyor et al. (2013).

#### 2.3.4.2 Short Base Line (SBL)

Contrary to the LBL architecture, this architecture does not require any seafloor mounted instruments which saves time and money. In this architecture, it is installed an array of three or more transducers on the hull of a ship or a surface platform. A transponder is attached to the vehicle or person in mission to be positioned.

One transducer sends an acoustic signal. The transponder attached to the submersible responds with another signal in a different frequency. The response is received by the transducer array, obtaining this way the two-way time of flight. The transducer array then computes the submersibles position using the trilateration method, explained in 2.3.6.

The systems level of precision is highly dependent on the spacing between the transducers in the array, the bigger the baseline the more precise the system becomes. Therefore, it is ideal that the transducers are placed as far apart from each other as possible, attached to the hull of a large vessel floating in the water or a dock. If the transceivers are far enough apart this system provides a level of accuracy comparable to LBL. Surveyor et al. (2013)

The range derived from an SBL system is significantly affected by the conditions on sight at the time of the mission. The ship/platform is floating on the sea surface and is subject to pitch, roll and raw movements. As such, an SBL system needs additional sensors such as a ship's gyro, a vertical reference unit (VRU), or sometimes a reference navigation system, to increase the derived range. Zhou (2010)

This technique is mainly used to track submersibles from a surface platform such as an oil drilling platform. It is also used to search crashed aeroplanes in the sea. Examples of commercial available SBL models are the SHARPS SBL system by Marine Sonic Technology and the RS5D SBL system by Nautronix. Zhou (2010)

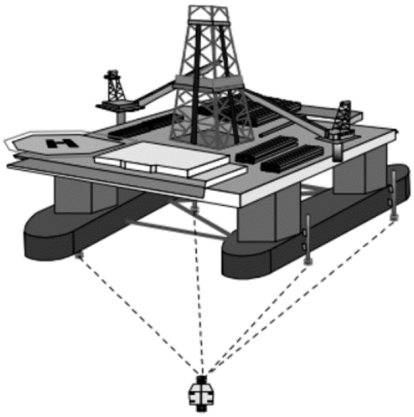


Figure 2.4: Short Base Line Rosa (2017)

#### 2.3.4.3 Ultra Shot Base Line (USBL)

This architecture is by far the most popular of the three presented. Tan et al. (2011) A USBL system relies on two primary elements, an array of transducers that are typically less than 10 cm apart Tan et al. (2011), (thus the ultra-short baseline), and a transponder.

The operation mode is very much similar to that of the SBL systems. A transducer sends an acoustic signal. Upon receiving this signal the transponder sends another one back in a different frequency. Once the different transponders receive the response, the round-trip time is known. Then, using the estimated sound speed, each one of the transducers calculates its distance to the transponder. The direction is calculated using the multilateration method explained in 2.3.6.

Figures 2.4 and 2.5 depict the difference between the SBL and the USBL architecture. In figure 2.4 the transducer array is spread as far apart as possible for maximum accuracy, whereas in figure 2.5 all transducers are spread across a small area. However, what this system gains in ease of use and deployment, it loses in the level of positioning accuracy. Tan et al. (2011)

Manufacturers of USBL systems include Nautronix, Sonardyne, IXSEA (GAPS pre-calibrated Ultra-Short BaseLine), Applied Acoustics (EASYTRAK USBL), LinkQuest (TrackLink USBL), Tritech (Micron Nav), Kongsberg (HiPAP - High Pre- cision Acoustic Positioning), and EvoLogics (USBL Acoustic Modem). Tan et al. (2011)

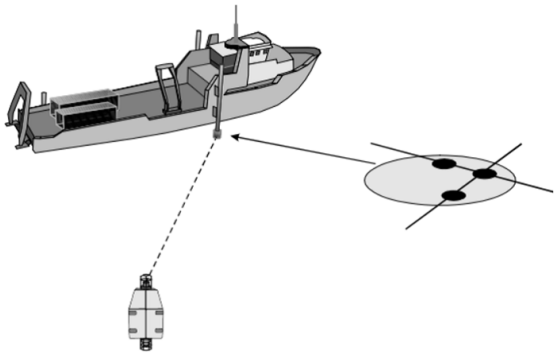


Figure 2.5: Ultra Short Base Line Example Rosa (2017)

### 2.3.5 Time Difference of Arrival (TDoA)

Time difference of arrival is the estimation of the time difference from the moment a signal reaches a receiving sensor, to the moment that same signal reaches a spatially separated sensor.

It is possible to determine the TDoA by first estimating the time of arrival at each sensor using one of the aproaches explained in 2.3.7, for example, using matched filters. However, even though a matched filter significantly reduces the effect of the accumulated noise facilitating the detection of the transmitted signal, there is a better approach.

In case the sensors involved are not too far apart, is correct to assume that the received signal by each sensor is roughly affected by the same noise. If that is the case, the signals received by the two sensors are very similar, and by extent, they have a high cross-correlation. For this reason, this is the most used approach.

For more information on the subject consult Chen et al. (2006), this paper focuses solely on this subject and proposes several adaptations to the cross-correlation method to improve the estimation.

### 2.3.6 Direction of Arrival (DoA)

In localisation systems is important to be able to determine the direction of the signal source, this is possible knowing the TDoA, the following are the two main strategies used:

#### 2.3.6.1 Multilateration

This technique requires two sensors, the two sensors are used to determine the TDoA, and from that, the direction of the signal is estimated.

Given TDoA, it's possible to create a hyperbola that describes all possible positions for the transmitter. This hyperbola is in itself not enough to determine the angle of the source. If the sound signals from the transmitter are plane, the direction of the source may be described by the asymptote of the hyperbola. The slope of the asymptote gives the angle between the sensors axis and the transmitter. "The difference between the hyperbola and the asymptote is that the

former assumes the waves to be spherical or circular and the latter assumes that the waves are plane." Dalskov and Olesen (2014)

Because the distance between receiver and transmitter is much greater than that between sensors at the receiver, it's acceptable to approximate the sound waves to plane waves. The equation of the hyperbola is found in equation 2.4:

$$y(x) = \frac{1}{2}\sqrt{\frac{(4x^2 - \Delta l^2)(l_{2,1}^2 - \Delta l^2)}{\Delta l^2}} \tag{2.4}$$

Where $l_{1,2}$ is the microphone separation and $\Delta l$ is the length corresponding to the TDoA:

$$\Delta l = TDoA \times c \tag{2.5}$$

Where c is the estimated sound speed at sight.

As explained before, what describes the angle of the signal relative to the microphones is the asymptote of the hyperbola, and this slope is described by the function:

$$y(x) = x\frac{\sqrt{l_{2,1}^2 - \Delta l^2}}{\Delta l} \tag{2.6}$$

Finally, the angle of the slope is given by:

$$\theta_a = \arctan\frac{\sqrt{l_{2,1}^2 - \Delta l^2}}{\Delta l} \tag{2.7}$$

Equations 2.4 through 2.7 were extracted from Dalskov and Olesen (2014).

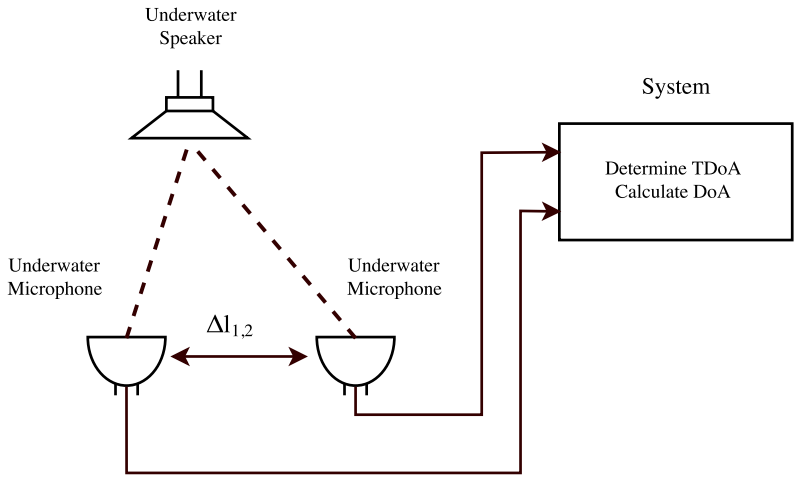



Figure 2.6: Direction of Arrival estimator

Using 2 microphones is possible to have the direction of the source in tow dimensions, The use of three microphones will render 3 hyperbolas and therefore allow for a 3-D localisation.

#### 2.3.6.2 Trilateration

Once again, this method depends on first determining TDoA, for simplicity reasons, the receiver in the configuration found in figure 2.7 uses two sensors, which are sufficient to determine the angle between the sensors axis and the transmitter on a two dimensional plane, one more sensor in the receiver cluster would allow three dimensional localisation.

The red point represents sensor number one, and the blue point represents sensor number two, the green point represents the transmitter. The distance between the transmitter and sensor #2 is R, the distance between the transmitter and the sensor #1 is (R+r), where r is equivalent to the distance a sound signal travels during TDoA.

The following three equations where deduced in Valente (2016).

$$X = \frac{r(r+2R)}{4d} \tag{2.8}$$

$$Y = \pm\frac{1}{4}\sqrt{\frac{-16d^4+8d^2r^2-r^4+16d^2rR-4r^3R+16d^2R^2-4r^2R^2}{d^2}} \tag{2.9}$$

$$\alpha = \arctan\frac{X}{Y} \tag{2.10}$$

All the parameters of the equations are depicted in figure 2.7.

Equations 2.8 through 2.10 allow us to determine $\alpha$ as represented in figure 2.7, $\alpha$ is 0 degrees when the transmitter is situated along the bisectrix created by the to sensors of the receiver. The result varies from -90 to 90 degrees from left to right. Using this setup is not possible to establish whether the source of the signal is situated in front or behind.

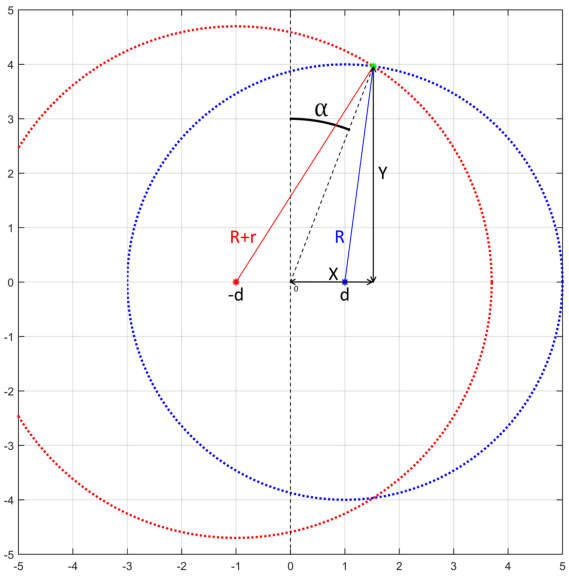



Figure 2.7: Trilateration method Valente (2016)

### 2.3.7 Time of Arrival

All the systems based on acoustic signals have one thing in common, they all need to determine the time of arrival of the signal to establish a position. Hence the importance of this subsection.

#### 2.3.7.1 Detection by Frequency Selective Amplification

The method here presented is the approach currently used in the AUVs of INESC Tec. The system is represented in figure 2.8.

The system uses a set of pure tone signals with different frequencies. Figure 2.8 shows a signal generator that when requested, generates a tone with one of the used frequencies. A transducer then transmits the generated signal.

The signal then reaches the hydrophone, and from the hydrophone is filtered. The system uses several pass-band filters working in parallel, each with the pass-band around one of the used frequencies.

The output of each filter is connected to an amplifier with an exaggeratedly high gain. A given amplifier will, therefore, saturate when the system is receiving a signal with the frequency allowed through by the band-pass filter.

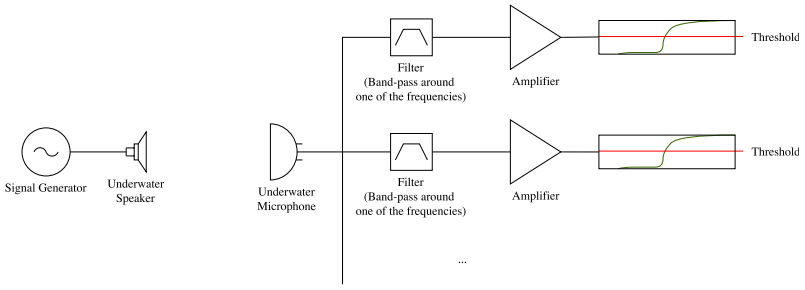



Figure 2.8: Detection by amplification of each used frequency

The time of arrival detection of the acoustic signal is acquired when the output of one amplifier reaches a predefined threshold.

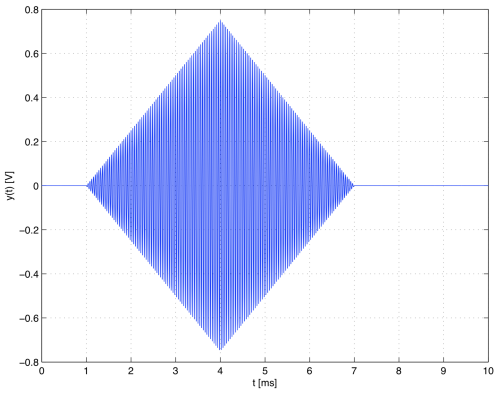
This technique produces satisfactory results. However, the system can be improved by using the correlation method explained ahead.

Another feature of this system is the ability to transfer data. Each frequency has associated with itself a pre-established command that the receiver must execute. Even though the bit rate is low, this feature is useful to control an AUV in a mission.

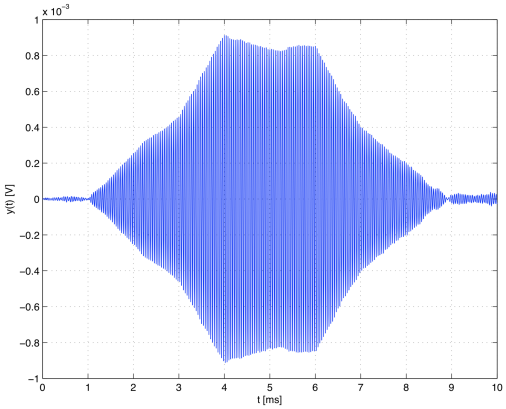
#### 2.3.7.2 Correlation Based Detection

This method consists in correlating the expected signal by the signal being received. However, the type of signals used influences the precision of the time of arrival estimation. Figure 2.9a, shows the correlation result obtained when the used signal is a pure tone. Figure 2.9b shows the result of

the correlation when the received signal has added multipath. When in the presence of distortion, the use of a pure tone does not present good results.




(a) Ideal Cross correlation Oliveira (2009)




(b) Cross correlation with noise Oliveira (2009)

Figure 2.9: Ideal vs real cross correlation

The most common approach is to used signals modulated by Pseudo-Noise (PN) sequences. As explained in Obara et al. (2002), the time of arrival estimation based on correlation is a very straight forward process. After the received signal is amplified, it is shifted to base band, filtered and correlated with the transmitted bit stream. When the transmitted signal is received, the result of the correlation forms a peak which indicates the time of arrival.

The work developed in Oliveira (2009) presents a system capable of determining the 3 dimensional position of the receiver relative to the transmitter. It uses 4 transducers strategically placed at the receiver and calculates its position based on the time difference of arrival between traducers. The system starts by determining the ToA at each transducer, to do this, it uses signals modulated by PN sequences. With the transmitter and receiver separated by approximately 70 meters, the calculated distance based on the ToA results yielded a standard deviation of 9 cm.

In Obara et al. (2002) was developed an analog circuit that implements the described process. The main goal of this work was to have a system capable of detecting a transmitted signal over large distances. Therefore, because the attenuation of acoustic signals increases inversely to the increase of the signals frequency, the channel frequency used was 264 hertz and the signal had a band-with of 3 Hz. The higher the duration of a signal the higher the energy of that signal, so, also to increase the chances of detecting the transmitted signals over large distances, were used signals with a duration of 32 seconds modulated by 1024 bits. The implemented system showed that it was capable of detecting signals transmitted at distances over 1000 km with a distance estimation error in the order of a few hundred meters.

In Huang et al. (2016) is proposed a method also based on the correlation of signals modulated by Pseudo-Noise (PN) sequences. However, this work proposes an alteration to the signals involved in the correlation. One of the main goals of this work was to develop a system significantly less expensive then one previously developed. This new system should be able to work with a cheap low power consumption piezzo-electrical cristal oscillator. This cheaper oscillator has an

unstable frequency, which results in transmitting signals with frequency deviations. In the previously presented works, the correlation consisted in correlating the received signal by a reference signal, the reference signal being the transmitted signal. This work proposes the use of dual PN sequences to estimate the time of arrival. In this setup, the transmitted signals consist in transmitting a modulated PN sequence followed by a guard time followed by the same PN sequence. The time of arrival estimation is derived from correlating the two PN sequences transmitted per signal. This idea was explored in an attempt to diminish the problems associated with the Doppler effect ,multipath and clock drift. In this setup, the signals involved in the correlation suffer roughly the same distortions. Therefore, the system should be more robust to adverse conditions. The presented results show that this setup is capable of yielding a correlation peak in conditions where the previously developed system based on a single PN sequence failed. However, no values relative to the precision of the system were presented.

# Chapter 3

# Functional Implementation

This chapter proposes an algorithm to improve the time of arrival estimation and describes the functional implementation of that algorithm. Matlab simulations are carried out to understand the viability of the algorithm. In the preliminary simulations were used ideal signals, then, the steps of the algorithm were validated using real recorded underwater signals. These signals were transmitted and recorded using the same hardware of the final implementation.

The final implementation runs on a Xilinx Zynq-700 which includes a Field Programmable Gate Array (FPGA) and an ARM processor. The clock frequency of the FPGA is 125 MHz, this frequency is set by the wrapper that integrates the implemented hardware module. The FPGA is connected to two analogue to digital converters and one of them is used to sample the signal from the hydrophone. The characteristics of the hardware are further analysed in chapter 4.

The signals used to develop the functional implementation had the same specifications as the ones used in the final implementation. So, in this phase, it was defined the sampling frequency, carrier frequency, bandwidth and the properties of the binary sequences. These parameters were chosen based on trade-offs among the characteristics of the hardware used, results of time of arrival yielded and the hardware complexity of implementation.

Some calculations conducted during this chapter depend on the underwater sound speed. The considered sound speed for the calculations was 1500 m/s.

## 3.1 Solution Method

As explained in 2.3.7, determining the exact moment at which an acoustic signal reaches an acoustic transducer is crucial to every localisation system with an architecture based on acoustic signals.

The work here developed is based on correlation, however, as explained before, the correlation technique is not an entirely reliable method by itself. When in the presence of noise and multipath effect, it can be difficult to obtain an accurate time of arrival from the resulting correlation peak

The standard deviation of the correlation is given by: Oliveira (2009)

$$\sigma_{TOA} \geq \frac{1}{BW\sqrt{\frac{2E}{n_0}}} \tag{3.1}$$

Where $BW$ is the bandwidth and:

$$\sqrt{\frac{2E}{n_0}} \tag{3.2}$$

is the SNR of the received signal. $E$ is the signal energy, and $n_0$ is the spectral density of the noise.

The objective of this work is to decrease the standard deviation of the correlation peak, and therefore, minimise the error when estimating the ToA. Based on equations 3.1 and 3.2, there are two methods by which is possible to decrease the standard deviation, the first one is by increasing the SNR, and the second one is by increasing the bandwidth of the signal.

The SNR can be increased by increasing the energy of the signal and by decreasing the noise of the environment. The noise depends on the environment and not much can be done to improve that apart from filtering unused frequencies. So the main thing to do to increase the SNR is to increase the energy of the signal.

The other option is to increase the bandwidth of the signal, to achieve that, were used signals modulated in BPSK by pseudo-random binary sequences. The modulation of pseudo-random binary sequences (PRBS), despite being deterministic, results in a signal with the properties of a random sequence. Oliveira (2009)

It is possible to generate a set of pseudo-random binary sequences that have a high auto-correlation and low cross-correlation, making it easy to identify which sequence is being received. So, using this property, it is possible to establish a low bit rate communication stream by attributing a symbol to each of the used pseudo-random binary sequences.

The number of pseudo-random binary sequences used affects length of the signal. Hardware characteristics dictate this parameter. The number of sequences will also determine the bit rate, where the number of bits per symbol is $\log_2 N$, (N is the number of pseudo-random binary sequences used). For reasons further explored ahead, it was used a set of 16 pseudo-random binary sequences.

Figure 3.1 illustrates the system to be developed. At the emitter end, signals are generated and transmitted. Each generated signal results from modulating one of 16 different pseudo-random binary sequences in BPSK.

At the receiver, the received signal is correlated in parallel by each of the 16 possible transmitted signals. Due to the cross-correlation properties of the used binary sequences, only one of the correlations will yield a correlation peak for a given signal, making the received signal identifiable.

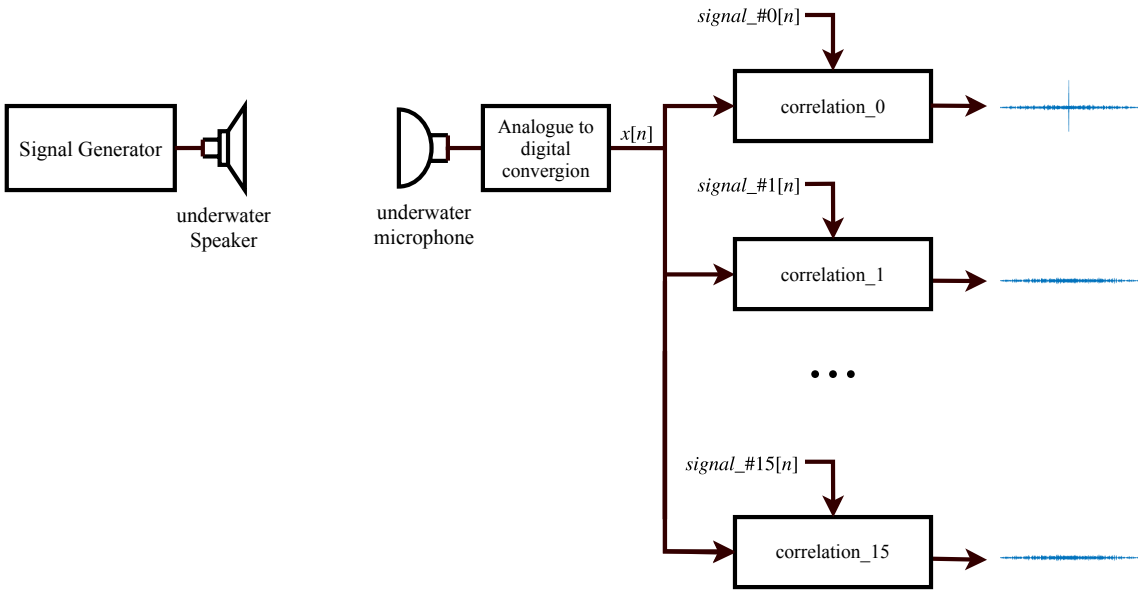



Figure 3.1: Systems Design

## 3.2 Hardware design constraints

In chapter 4, the hardware used in the system is presented in detail. Here, only the specifications that constraint the functional implementation are presented

The analogue to digital converters have a maximum sampling frequency of 125 MHz, given that there is a correlation result per sample, receiving samples at 125 MHz would translate to a time resolution between samples that results in having a maximum precision of:

$$\frac{sound\_speed}{sampling\_frequency} = 0.012\,\text{mm} \tag{3.3}$$

However, receiving samples at 125 MHz, would require the system to calculate 16 full correlations per clock, which is very resource consuming. Therefore, there is a need to establish a trade-off between the required precision, and the opportunity to implement modules with sequential logic. Sequential logic is essential for a good use of the available resources. The system decimates the samples by 512 which results in a sampling frequency of:

$$\frac{125}{512} \approx 244.141\,\text{kHz} \tag{3.4}$$

and renders a maximum precision of:

$$\frac{sound\_speed}{new\_sampling\_frequency} = 6.144\,\text{mm} \tag{3.5}$$

A decimation of 512 allows the implementation to use 512 clock cycles to calculate the correlations. The reasons to settle on this specific value are further discussed in 4.

The hardware used in the transmission of the signals also determines which channel frequency to use and the available bandwidth. The next chapter explores these limitations. The functional implementation was developed considering that the available bandwidth is situated approximately between 20 kHz and 30 kHz.

## 3.3 Signal modulated by a PRBS versus pure tone

Correlation is the method of choice to determine the time of arrival. A correlation peak appears when the received samples are most similar to the expected, however, the type of used signals can yield a more or less sharp correlation peak. The use of a pure tone generates the correlation result presented in 3.2 left plot. The correlation value starts to increase linearly from the moment the first sample of the signal is received. The value of the slope created by the correlation peak can be calculated by:

$$\frac{\sum_{n=1}^{n\_signal\_samples} y(n)^2}{n\_signal\_samples} \times sampling\_frequency \tag{3.6}$$

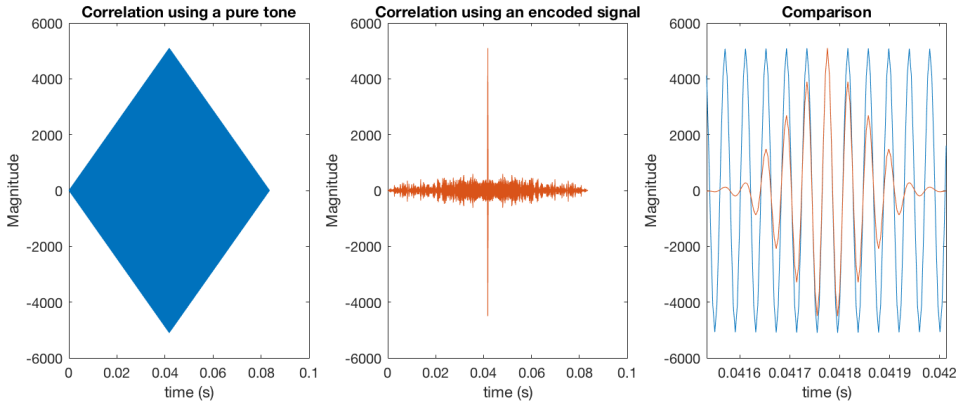



Figure 3.2: Correlation result, pure ton versus encoded signal

That same figure 3.2, centre plot, shows the result of the correlation when using signals modulated in BPSK by a PRBS. As all graphs in the figure have the same scale, it is evident the difference in the sharpness of the correlation peak. These results are obtained using ideal signals, so they both present a correct result, however, having a sharper correlation peak produces better results when in the presence of noise, multipath and Doppler.

As previously stated, the available bandwidth is situated roughly between the 20 and the 30 kHz, for a good use of the available bandwidth, the carrier used has a frequency of:

$$\frac{fpga\_clock\_frequency}{decimation\_factor \times 10} \approx 24.414\,\text{kHz} \tag{3.7}$$

This frequency is chosen because it is situated half-way between the available bandwidth, and also because a period of this signal is sampled by an integer number of samples at the receiver, which simplifies significantly the signal processing system.

In BPSK, the information is in the phase of the signal, phase $\phi$ represents 1 phase $\phi + 180$ represents a 0, as depicted in figure 3.3.

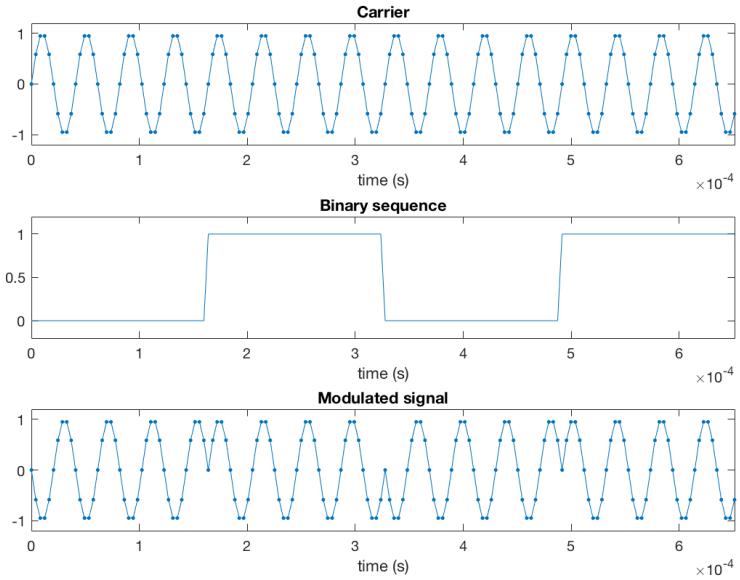



Figure 3.3: BPSK Modulation

The main lobe in the spectrum of a signal modulated in BPSK spreads across:

$$\frac{2}{T} \tag{3.8}$$

where T is the period of each bit. The used signals modulate a bit every four periods of the carrier frequency, that translates to a bandwidth of:

$$\frac{2}{carrier\_period \times 4} = 12.207\,\text{kHz} \tag{3.9}$$

which results in a use of the spectrum by the main lobe from approximately 18.311 kHz to 30.518 kHz. This way, all the available spectrum is used and each bit is modulated by an integer number of periods of the carrier frequency. This is exploited to simplify significantly the signal processing system.

Figure 3.4 represents the resulting spectrum from the proposed modulation.

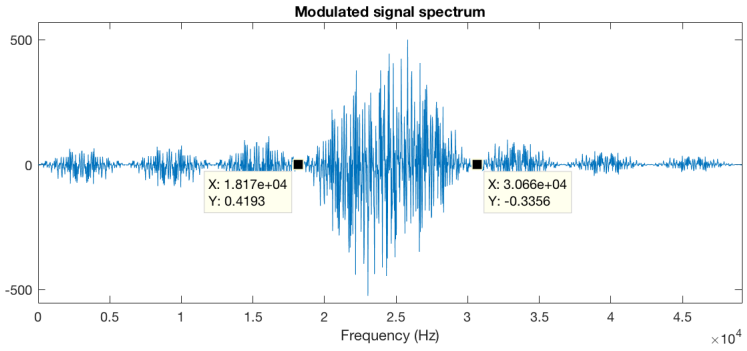



Figure 3.4: Spectrum of used signals

The use of modulated signals with the stated characteristics yields the correlation result depicted in figure 3.2, centre plot. The intensity of the peak is the same as if a pure tone with the same duration and amplitude is used. However, the peak is a lot sharper. The reason for this sharper peak is that the correlation value only starts to increase once the bits of the received signal and the bits of the reference signal used in the correlation are all aligned, and that only happens when the last 4 periods of the signal start being received, (each bit modulates 4 periods of the carrier). So, under these conditions, the slope of the correlation becomes:

$$\frac{\sum_{n=1}^{n\_signal\_samples} y(n)^2}{n\_samples\_per\_period \times 4} \times sampling\_frequency \tag{3.10}$$

## 3.4 Pseudo-Random binary sequences

This section defines the set of pseudo-random binary sequences used. Trade-offs need to be established between the number of sequences used and their length. These specifications affect the length of the modulated signal, and by extent, affect the energy spent by the transmitter, and the resources necessary to perform the correlation at the receiver.

As explained before, the system takes advantage of the possibility to generate a set of pseudo-random binary sequences with low cross-correlation, this way, data can be transferred from transmitter to receiver by just assigning a symbol of n bits to each sequence.

The use of Walsh and Hadamard codes were considered as well as Gold and Kasami sequences. The first two originate orthogonal sets of binary sequences. However, the autocorrelation results have several peaks instead of a single narrow correlation peak, which in the presence of noise can yield wrong time of arrival estimations. Kasami and Gold sequences can generate larger sets of binary sequences, and the auto-correlation produces a single peak. However, these binary sequences are not orthogonal between them. The auto and cross-correlation results are from the following group $\{-1,\ -1 \pm 2^{\frac{n}{2}},\ -1 \pm 2^{\frac{n}{2}} + 1\}$. The compromise is to use Kasami sequences of the small set. These sequences, yield auto and cross-correlation results from the following group $\{-1,\ -(2^{\frac{n}{2}} + 1),\ 2^{\frac{n}{2}} - 1\}$. Contrary to the sequences of a large set, the cross-correlation results for the small set of kasami sequences satisfies the Welch lower bound. Dinan and Jabbari (1998).

Therefore, by using a small set, there is less risk of making a mistake when identifying the received sequence.

The number of data bits carried by each sequence is $\log_2(\#sequences)$, and the number of sequences one can generate from a generator polynomial of a small Kasami set is:

$$\#sequences = 2^{\frac{d}{2}} \tag{3.11}$$

Where $d$, is the degree of the used polynomial. Ideally, the set of sequences would be as large as possible to transmit more data bits per signal. However, to have a broader set of sequences, the degree of the polynomial generator needs to increase, and with it, the length of each binary sequence.

$$binary\_seq\_length = 2^d \tag{3.12}$$

The used generator polynomial is:

$$p(z) = z^8 + z^4 + z^3 + z^2 + z^0 \tag{3.13}$$

Which translates to using a set of 16 binary sequences with 255 bits. Having a set of 16 binary sequences equates to transmitting 4 data bits per transmitted signal.

Section 3.3 defines that each bit of the pseudo-random binary sequence modulates 4 periods of the carrier, therefore, each modulated signal is represented by 10200 samples at the receiver.

## 3.5 Correlation Method

The correlation is performed directly with the samples in channel frequency, shifting the signal to base-band would present an advantage if downsampling below

$$2 \times upper\_frequency\_of\_main\_lobe \approx 61.036\,\text{kHz} \tag{3.14}$$

was necessary, to not incur in aliasing. As previously stated, the samples are processed at 244.144 kHz, for good precision. Therefore, there is no need to shift the spectrum to base-band.

Sections 3.3 and 3.4 define that the system uses 16 pseudo-random binary sequences modulated in BPSK and that each signal is represented by 10200 samples at the receiver. It is also defined that two consecutive samples are received every 512 clock cycles and that there is a correlation result per sample for each of the 16 signals. These stipulations require the signal processing digital platform to process:

$$\#sequences \times \frac{\#samples\_per\_signal}{\#clock\_cyles} \approx 320 \tag{3.15}$$

multiplications and additions per clock cycle, which results in 80 Gs operations. These many computations would use a large portion of resources of the hardware that runs the implementation of the system. This section presents a simplification of the correlation method to save resources.

A correlation consists of multiplying each sample of the received signal by each sample of a reference signal which is the expected signal. A peak will arise when those signals best match. In the case of the signals in question, a peak will appear when both, the received signal and the reference signal, at every 4 periods, have the same phase. When this happens, in ideal conditions, a positive received sample is multiplied by a positive reference sample and a negative received sample is multiplied by a negative reference sample. All does multiplications have a positive magnitude, and added, form the maximum of the correlation.

Because BPSK is used, all the information is in the phase of the signal. Considering this, instead of multiplying the received samples by exactly the samples of the reference signal, if the received samples are multiplied by 1 when the reference sample is positive and $-1$ when the reference sample is negative, the correlation peak appears exactly at the same time, with only a difference in amplitude. This way, there is no need to have 320 parallel multipliers, just a condition that adds or subtracts the sample to the correlation accumulator.

Figure 3.5 illustrates the principle. The top left plot is the reference signal, which means, this is the received signal in ideal conditions, and therefore, the signal one would use in a typical correlation. The blue part represents one bit, the red part represents a second bit. The top right plot is the result of correlating the reference signal by itself, it is therefore, the result of an exact correlation. The bottom left plot is a square signal that represents adding or subtracting the sample being correlated with. The bottom right plot is the result of correlating the received signal by the square signal. This last correlation applies the proposed method, only adds or subtracts a sample. As expected, the peak appears at the same time, and that is the only requisite of the correlation.

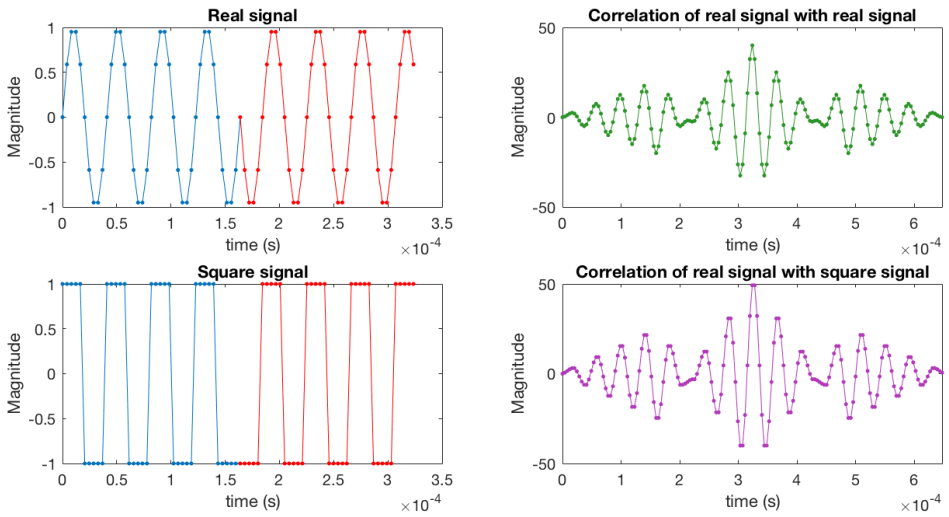



Figure 3.5: Correlation using ideal versus square signal

## 3.6 Offline validation using real recorded signal

This section presents the results of the correlation performed offline on a real recorded signal. These preliminary tests were carried in a 4 by 4 by 1.6 meter laboratory test tank. This is an adverse environment, in it, there is no noise interfering with the experiment but, because it is such a confined space, the record is severally affected by multipath due to reflections in the walls bottom and top.

Figure 3.6 top plot, shows the correlation result for the recorded signal. Three different signals were transmitted and received during the recording. The recording is correlated with a reference signal for one of the 16 binary sequences. Because three different encoded signals were transmitted, only one originated a clear correlation peak, as seen in the same figure.

Figure 3.6 also compares the result of the correlation when correlating the received signal by the real modulated signal versus when correlating using the proposed method. The correlation peak yields the same time of arrival in both approaches, only with a difference in magnitude.

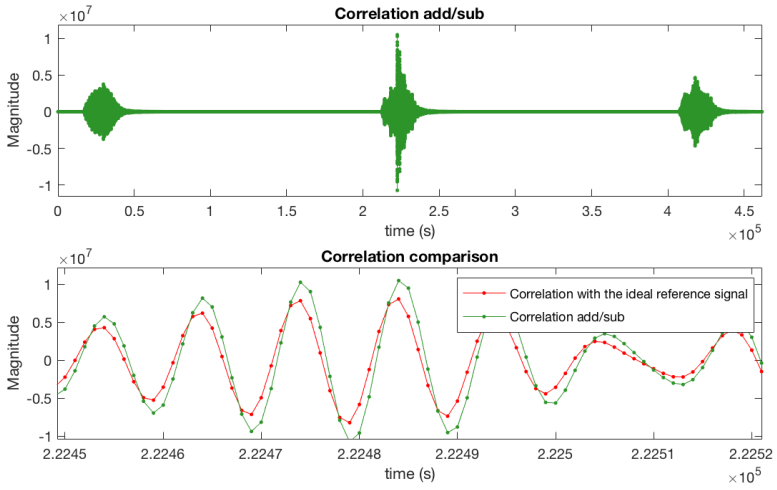



Figure 3.6: Exact correlation versus proposed method

## 3.7 Correlation Peak Improvement

When the received signal and the reference signal used in the correlation start to align, the correlation result starts to increase, and local maximums and minimums surround the absolute maximum of the correlation. This is represented in figure 3.7. The top plot is the result of the correlation. The central plot is a closer look at the correlation peak, in this plot, the maximums and minimums that surround the absolute maximum are evident.

In real conditions, the received signal has added noise, frequency deviations due to Doppler and added reflections due to multipath. The maximum of the correlation under these circumstances may not represent exactly the moment a signal has arrived. In the worst case, the local peaks to the right and the left may even be higher than the peak that represents the arrival of the signal.

Two local maximums are 10 correlation results apart, which means, the error when choosing the wrong peak is:

$$\frac{10 \times sound\_speed}{sampling\_frequency} \approx 6.14\,\text{cm} \tag{3.16}$$

This section presents a simple technique to improve the ToA estimation. This technique takes advantage of the fact that the local maximums and the module of the local minimums should form a perfect triangle in ideal conditions. These points are highlighted with green stars in figure 3.7 bottom plot. The idea is to extract the coordinates of these points and approximate each side of the correlation peak to linear equations, the intersection of the two linear approximations should represent a more exact ToA. The red cross in the same plot represents the intersection.

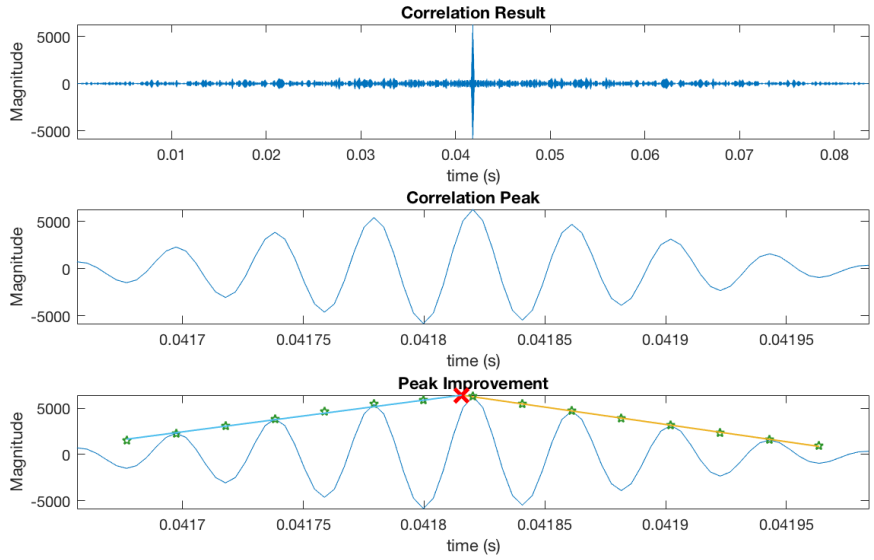



Figure 3.7: Correlation peak interpolation

## 3.8 Multipath

As stated before, these preliminary tests were conducted in a confined adverse environment for multipath. Figure 3.8 shows the correlation result of a real recorded signal with added multipath. The figure has two labels, each pointing a peak of the correlation. The one on the left is pointing to the correlation peak resulting from the arrival of the direct signal. The arrow on the right points to a correlation peak resulting from multipath. The correlation peak on the right has the higher magnitude of the two. This is possible if several reflected signals arrive at the hydrophone in phase.

From the several tests performed, these correlation peaks resulting from multipath are generally not higher than two times the value of the correlation peak that occurs when the direct signal arrives. Therefore, a rule can be applied to only acknowledges a new peak, when that peak has

a magnitude more than two times higher than any previous peak. (Excluding the 40 points that precede a maximum, local peaks appear in these 40 points because bits start to align).

This approach improved the percentage of correct results in the laboratory testing environment. However, not enough tests were carried out to generalise the method to all environments.

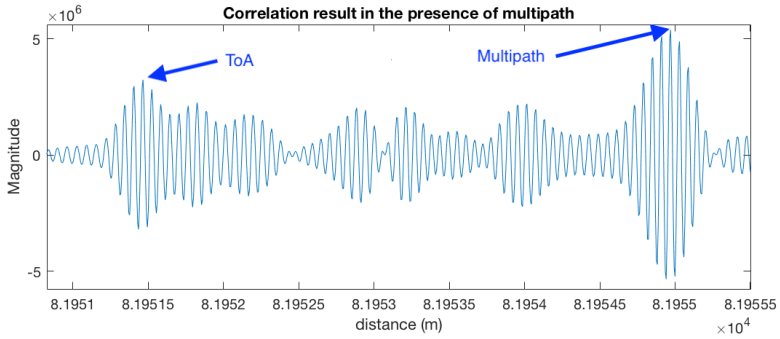



Figure 3.8: Record and correlation result

## 3.9 Doppler Compensation

Doppler effect occurs when transmitter and receiver move relative to each other. The frequency deviation as a function of the relative speed between transmitter and receiver can be calculated by:

$$Frequency\_deviation = \frac{relative\_speed}{sound\_speed} \times signal\_frequency \tag{3.17}$$

Matlab simulations where performed to study the impact of the Doppler effect. The signals used in these simulations have the same characteristics as the ones used in the actual system: same pseudo-random binary sequences, signal frequency and sampling frequency.

First, was assessed how much Doppler effect the system can tolerate without any compensation. Figure 3.9 illustrates the result of this simulation. The top plot shows the result of the correlation when the ideal signal is received. The centre plot shows the correlation result when the received signal has a 20 Hz frequency deviation from the expected carrier frequency. In this case, the correlation peak of the signal is significantly attenuated, approximately 6 times lower. However, the peak is clear, and the error concerning the estimated distance is below 1 centimetre. The bottom plot shows the result of the correlation when the received signal has a frequency deviation of 25 Hz. In this plot, no evident peak can be identified.

Based on this test, it is considered that the frequency deviation needs to be under 20 Hz for the system to function correctly.

However, tolerating only a frequency deviation of 20 Hz means that the relative speed between transmitter and receiver can not be higher than:

$$\frac{20 \times sound\_speed}{signal\_frequency} = 1.2\,\mathrm{m/s} \tag{3.18}$$

Generally, autonomous underwater vehicles may move at higher speeds. Speeds in the order of the 3 m/s. Thus, tolerating only 20 Hz of frequency deviation is not acceptable.

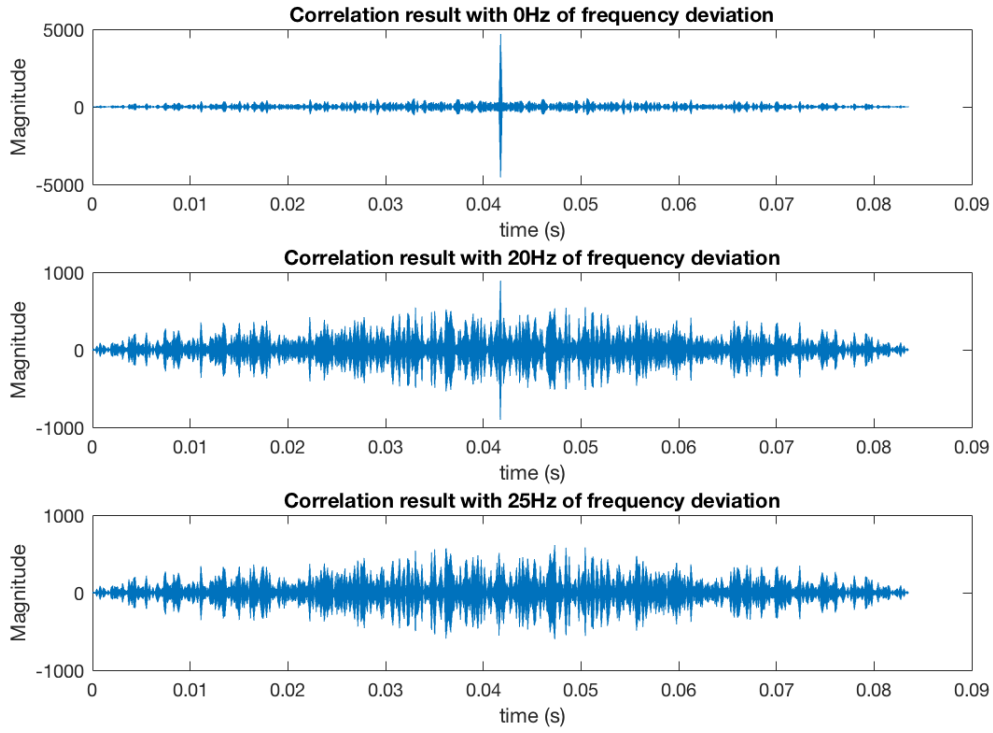



Figure 3.9: Correlation result when in presence of Doppler

The process to compensate for the frequency deviation caused by Doppler here presented consists of adding or removing samples to the received signal. This technique is used because it considerably reduces the implementation complexity of the receiver, this process can be implemented by merely skipping or buffering samples during the correlation.

Figure 3.10 depicts an attempt to compensate for Doppler effect. In this particular example, the frequency deviation is negative. Therefore, the full signal takes more samples to represent. To compensate Doppler, samples need to be removed, if it was positive a frequency deviation, samples would be added. Notice that each period of the signals is represented with a different colour. The top plot of the figure shows the ideal signal. The second plot shows the same signal with a 10% frequency deviation, each period goes from being represented by 10 samples to being represented by eleven samples. The algorithm consists in removing that extra sample. The result is shown in the third plot of the of the same figure. The bottom plot shows a comparison between the original ideal signal and the signal affected by Doppler after said Doppler has been compensated.

As shown, the algorithm consists of removing or adding a percentage of samples equally spaced along time, equal to the percentage of frequency deviation. This way, every time a sample

is removed or added, the phase of the signal affected by Doppler is aligned with the reference signal, as depicted in figure 3.10 bottom plot. Because BPSK is used, and therefore the amplitude of the samples doesn't carry information, only the samples close the magnitude level 0 create errors in the correlation. By the addition and subtraction correlation algorithm already presented, the difference that remains between the compensated signal and the original signal, creates a significant difference of intensity in the correlation peak, but, an insignificant difference concerning the ToA estimation.

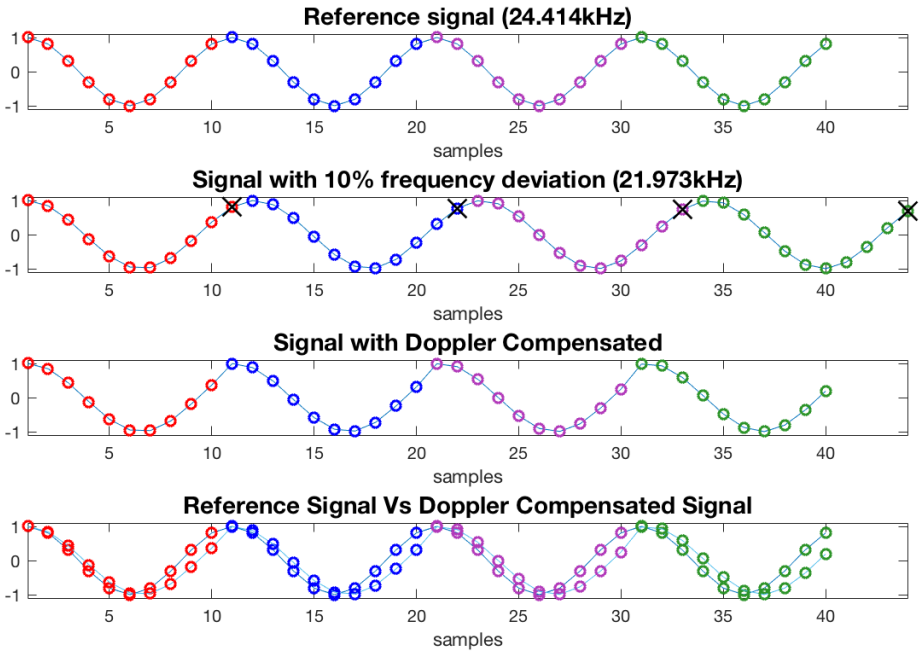



Figure 3.10: Doppler Compensation Process

Matlab experiments were carried out to understand the validity of the proposed solution to compensate for the Doppler effect. Figure 3.11 illustrates the results of the experiment. The top plot shows the result of the correlation when the received signal is not affected by Doppler. The second plot shows the result of the correlation when the received signal has 200 Hz of frequency deviation, as expected, there is no evident correlation peak. The third plot shows the result of the correlation of the received signal with a 200 Hz frequency deviation, after the Doppler effect has been compensated. The received signal has a frequency deviation of 200 Hz, which equates to $(200/carrier\_frequency) \times 100 = 0.82\%$. Following the proposed solution, one out of every $1/0.0082 \approx 120$ samples must be removed. The fourth plot shows a comparison between the result of the correlation for the ideal signal and the signal affected by Doppler after the compensation. There is a small difference in magnitude and no difference in ToA estimation.

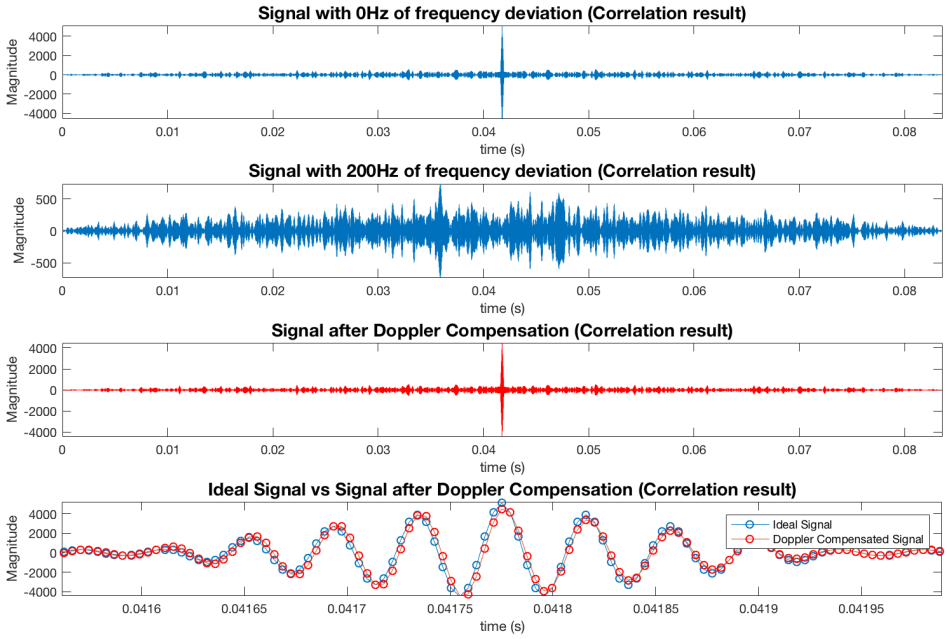



Figure 3.11: Doppler Compensation

As previously stated, the typical maximum speed of an AUV is in the range of 3 m/s. This speed creates a frequency deviation of 48.83 Hz. Therefore, the aim is to have a system resistant to this much Doppler effect.

It is shown previously that the system can tolerate up to 20 Hz of frequency deviation without any method of compensation, which means, there is a window of 40 Hz within which the system works. If two more windows of 40 Hz were added to each side of the spectrum, the system would be able to support frequency deviations from $\pm 60$ Hz. These values mean that the system would tolerate a relative speed of $\pm 3.6864\,\mathrm{m/s}$. The two needed extra windows are created by always compensating frequency deviations at $\pm 40$ Hz. If the system is compensating a frequency deviations of 40 Hz then it becomes immune to frequency deviations between 20 and 60 Hz, the same applies to compensating at $-40$ Hz.

Figure 3.12 depicts the results of the carried out Matlab experiments. The first plot shows the result of the correlation when the received signal is not affected by Doppler. The second plot shows the result of the correlation when the received signal has a 60 Hz frequency deviation. The third plot shows the result of the correlation of a signal with a frequency deviation of 60 Hz compensated at 40 Hz, as expected, the correlation presents a clear correlation peak. The fourth plot shows the difference between the first and the third plot. The error regarding distance is only of 1 cm.

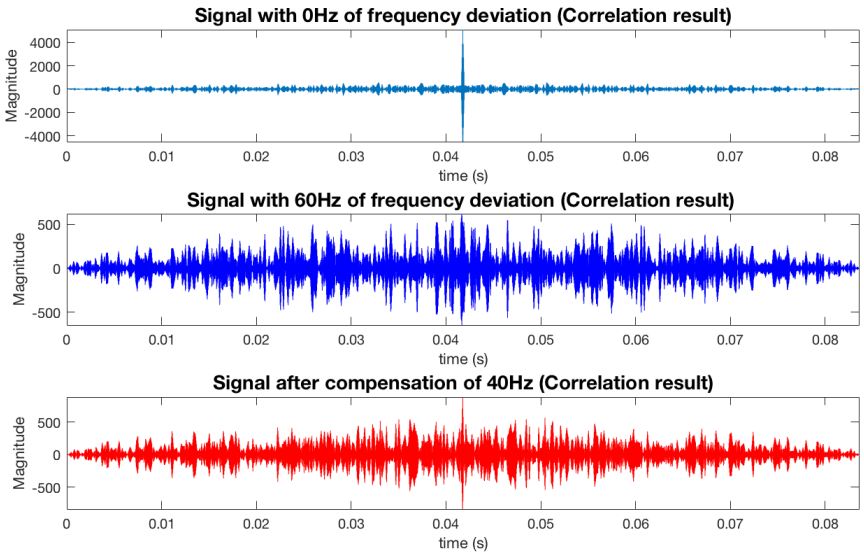



Figure 3.12: Maximum Doppler the system can support

Figure 3.13 shows two plots with results of the proposed algorithm applied to real recorded signals. The top plot shows the correlation result of a signal with a positive frequency deviation of 40 Hz, and the bottom plot shows the result for the opposite case. In both cases, three different signals out of the possible 16 were received. That is why only the second signal originated a clear correlation peak.

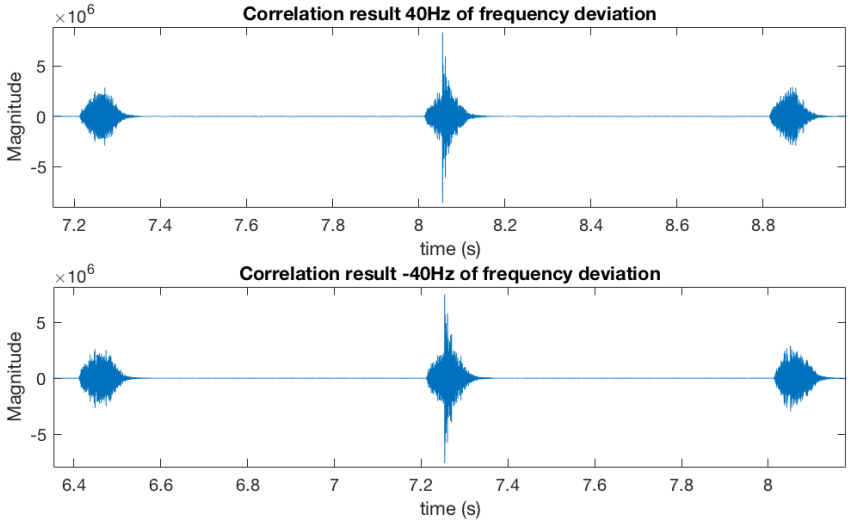



Figure 3.13: Doppler compensation with real signals

# Chapter 4

# Implementation

This chapter starts by presenting the hardware used in the implementation. Then, it describes the digital implementation and the developed software.

The following parameters of the used signals were defined in the previous chapter:

- The final sampling frequency is 241.44 kHz;
- The frequency of the carrier is 24.144 kHz;
- The system uses 16 binary sequences of 255 bits each;
- Each bit of the binary sequence, modulates in BPSK, 4 periods of the carrier.

These are the relevant parameters for the implementation. These parameters result in using signals with a duration of 41.8 ms which take 10200 samples at 241.44 kHz to be represented by the receiver.

## 4.1 Hardware platforms

This section presents the used hardware and its characteristics.

### 4.1.1 Transmission

The transmitted signals are generated by an FPGA, the output of the FPGA is connected to a signal amplifier that drives a piezoelectric acoustic transducer.

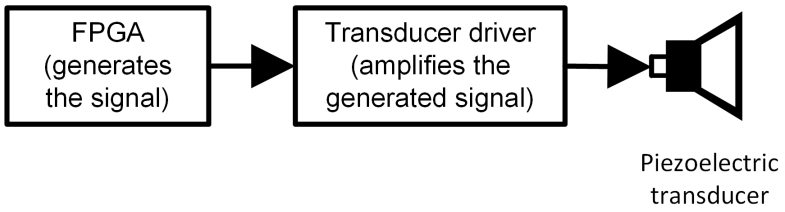



Figure 4.1: Hardware used in the transmission

#### 4.1.1.1 Digital platform

The board used for the modulation was an Atlys, which includes a Xilinx LX45 FPGA, the board is shown in figure 4.2. This FPGA has one 12-pin PMOD port used to output the generated signal.

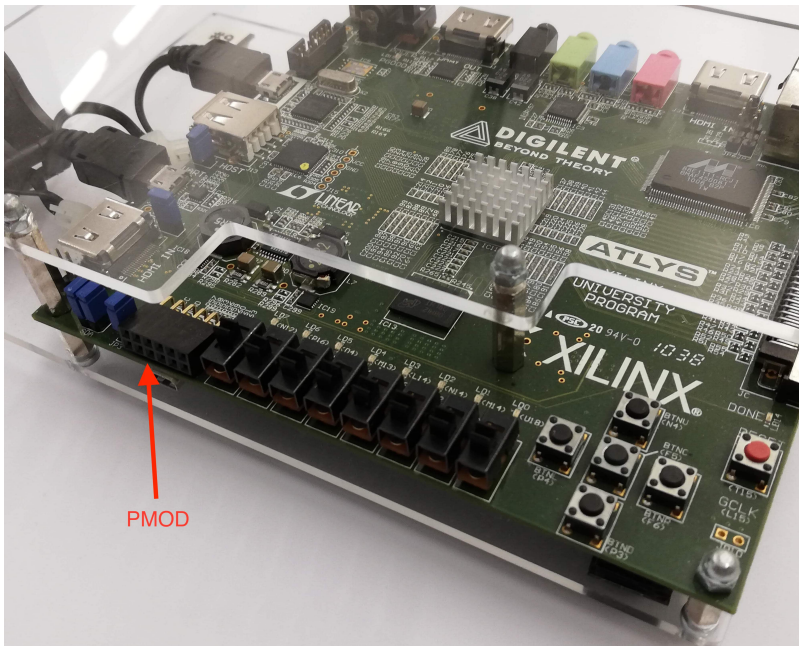



Figure 4.2: Atlys, used to generate the signals

#### 4.1.1.2 Transducer driver

Figure 4.3 shows the board used to drive the transmission device. The schematic of the board is in figure 4.4. The board is powered by 12 V and receives signals of 3.3 V.

The schematic, along with other components, shows two transistors and a transformer. The two transistors are driven by two opposite logical values and allow current to flow through the transformer in opposite directions. This makes it possible to generate a signal at the output of the transformer by alternating the logical values of gates of the transistors.

The large capacitance, labeled C2 in the schematic, is used as a power source. It charges when no signal is being transmitted and discharges when high surges of power are requested during transmission. This avoids requesting high peaks of current from the boards 12 V power source.

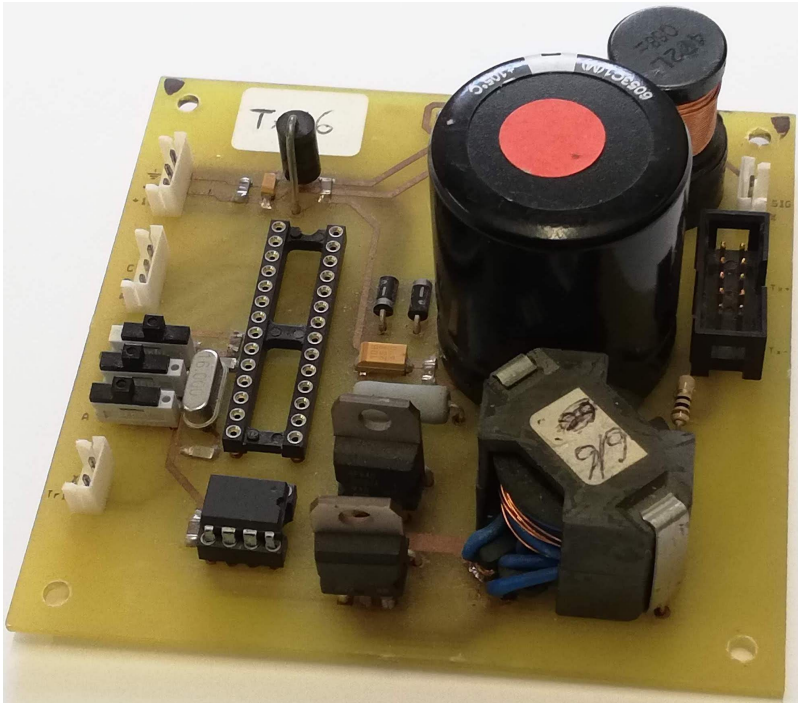


Figure 4.3: Board that drives the transducer

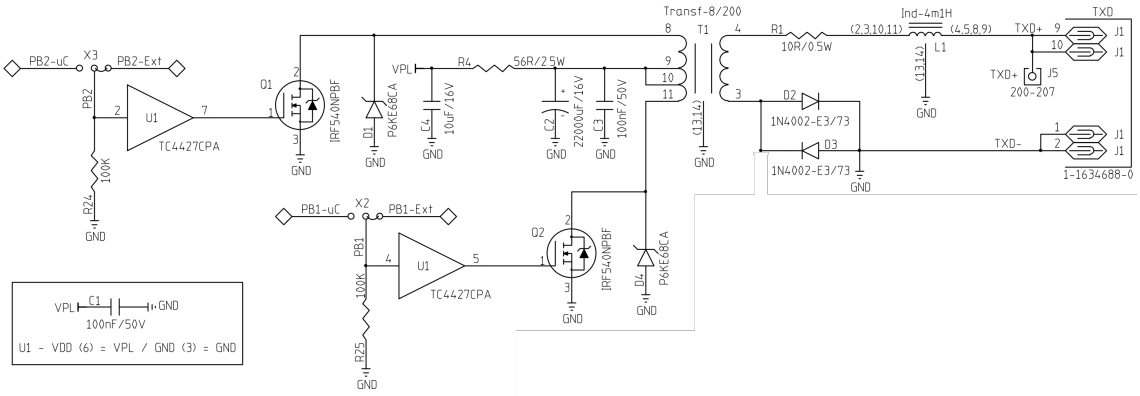



Figure 4.4: Schematics of the board that drives the transducer Cruz (2007)

Figure 4.5 shows the square signal that drives one of the transistors of the board. The other transistor is driven by the opposite logical value. The signal in orange is the output signal that drives the transducer. These oscilloscope plots where obtained with the transducer connected to the board. That is why the output of the board is a sinusoidal and not a square signal, the transducer acts as a filter. When the transistors are driven by square signal at 24.144 kHz, the peak to peak amplitude of the output is situated in a range between 40 to 60 V.

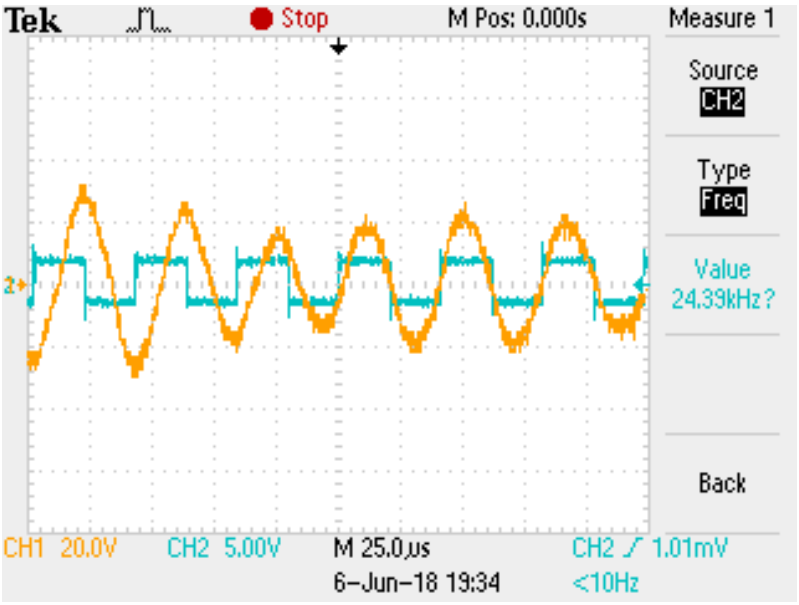



Figure 4.5: Transducer signal driver, oscilloscope image

#### 4.1.1.3 Transmission device

To transmit the acoustic signals it was used a transducer with the reference T217, from the company Neptune Sonar, Ltd. The used transducer transmits an acoustic signal when driven by an electric signal, and it is also possible to use it as a hydrophone. However, that feature was not used in this work.

This transducer is designed to work in frequencies between the 16 kHz and 30 kHz. It can transmit a maximum power of 400 W with a practically hemispheric pattern.

Figure 4.6: Transducer

### 4.1.2 Reception

The signals are received by an hydrophone, the hydrophone is connected to an analogue front-end that filters and amplifies the signal. The output of the analogue front-end is connect to an analogue to digital converter of a RedPitaya board, where the signal processing is preformed.

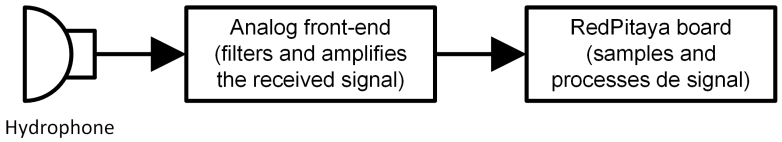



Figure 4.7: Hardware used in the reception

#### 4.1.2.1 Recording device

It was used an omnidirectional hydrophone to acquire the transmitted acoustic signals. The used hydrophone is shown in figure 4.8, is produced by the company *Aquarium audio products*, and the model is H2a hydrophone. The interface of the device is a 3.5 mm jack connector. The working frequency range of the hydrophone is situated between the 20 Hz and the 100 kHz.

Figure 4.8: Hydrophone

#### 4.1.2.2 Analogue front-end

Figure 4.9 shows the block diagram of the analogue front-end used between the hydrophone and the digital platform that performs the signal processing. Figure 4.10 shows the board on the right. As depicted in the block diagram, the analogue front-end has two variable amplifiers followed by an anti-aliasing low-pass filter.

The first amplifier of the chain is manually adjustable. The second amplifier represents two digitally controllable amplifiers, with an overall gain that varies from $0.1X$ to $2500X$.

The low-pass is a fourth order filter with the cut-off frequency roughly situated in the 250 kHz.

The input interface is a female 3.5 mm jack, the output is a female SMA jack. The board has two of these circuits, one for each channel. In this work, only one channel is used.

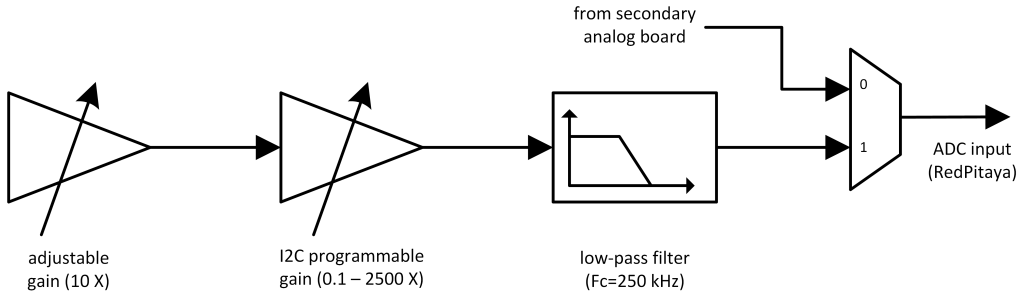



Figure 4.9: Analog front end

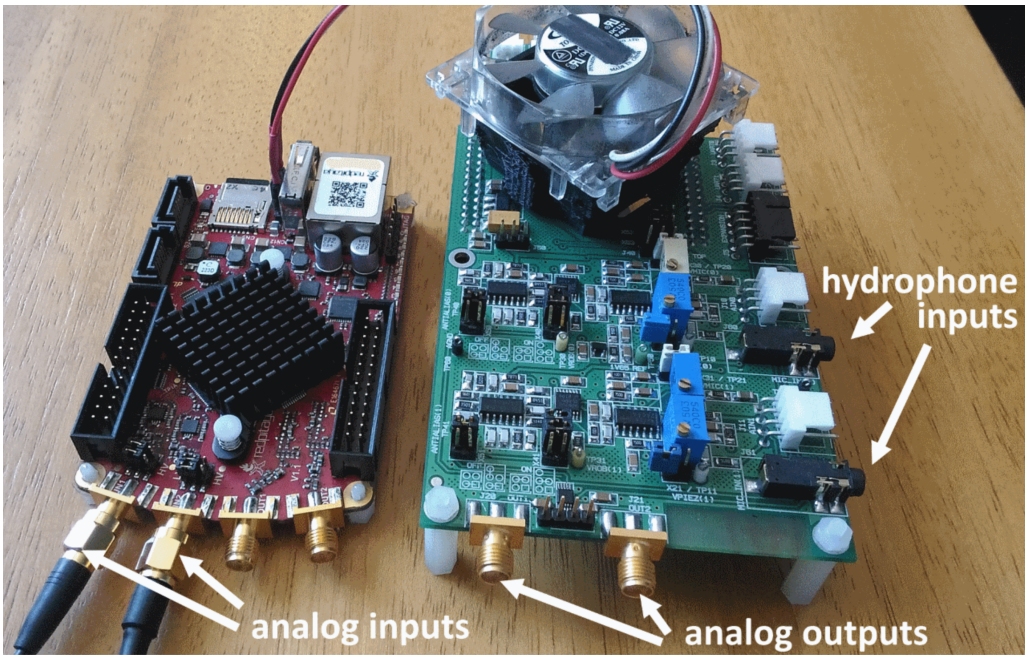



Figure 4.10: RedPitaya embedded computer (left), analog front-end board right Valente (2016)

#### 4.1.2.3 Digital platform

The digital platform used to perform the signal processing is the RedPitaya single board computer, shown in figure 4.10, left side. This board integrates a Zynq device that includes a dual-core ARM Cortex A9 processor and an FPGA.

The ARM processor runs Linux, uses DDR3 RAM, has an Ethernet interface, SD flash memory and three USB interfaces. This hardware is sold as a all-in-one multipurpose laboratory instrument, provides through a web interface functions such as an oscilloscope and a spectrum analyzer.

Two ADC's that sample at 125 Msamples/s and two DAC's are connected to the FPGA.

There is an implemented communication channel between the FPGA and the ARM processor. The provided open source Verilog project is used as a wrapper for the developed system.

## 4.2 Digital implementation

This section presents a description of the digital implementation of the transmitter and the receiver.

### 4.2.1 Transmitter

As previously stated, the digital device that generates the signals is an FPGA. The block diagram of the implementation is represented in figure 4.11. The transmitter implements a ROM that holds the 16 Kasami sequences. A signal is generated and outputted every time a trigger is received on the PMOD 3 of the FPGA.

As depicted in figure 4.11, when a trigger is received, a binary sequence is loaded from the ROM to a register. The least significant bit of the register determines the phase of the signal to transmit. The two possible signals are represented in figure 4.11. When the four periods corresponding to a bit are outputted, the register that holds the sequence is shifted one bit to the right, and the now least significant bit dictates which of the signals to transmit next. This process continues until the 255 bits are modulated. The output is set on PMOD 0. The logical opposite, necessary to control the second transistor of the transducer driver, is on PMOD 4. A period of the signal takes 512 periods to transmit because the FPGA is running with a clock frequency of 12.5 MHz.

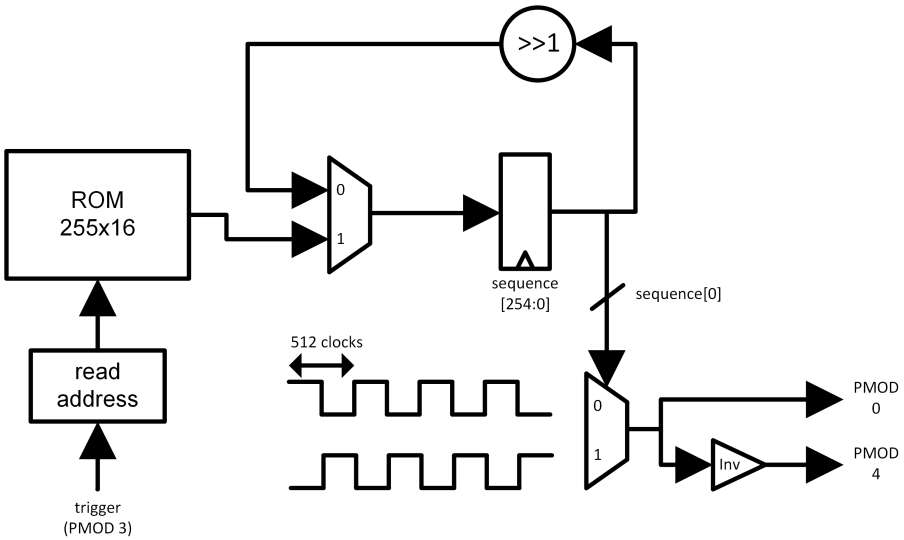



Figure 4.11: Block diagram of the transmitter implementation

### 4.2.2 Receiver

#### 4.2.2.1 Wrapper overview

Figure 4.12 shows how the developed system is integrated with the provided wrapper. The provided digital design implements other blocks, in this figure only the relevant ones are represented. As shown, the developed DSP receives samples from the provided decimator. Also provided, is

a communication interface between FPGA and the ARM processor. The ARM processor sets the parameters of the DSP, receives the results of the processed samples in real time, and runs an algorithm that executes further processing.

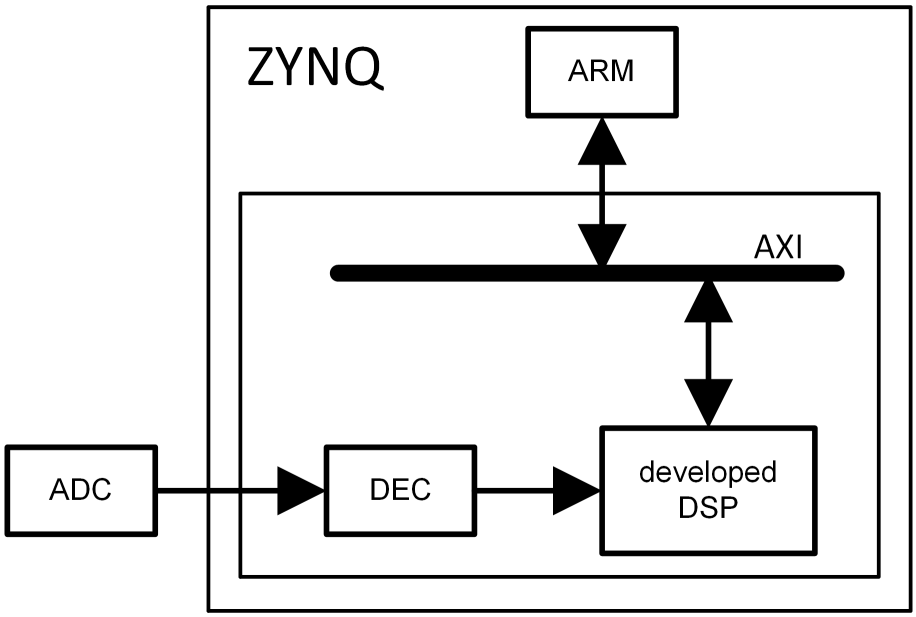



Figure 4.12: Block diagram of the digital system wrapper

#### 4.2.2.2 Top-level overview

The block diagram of the top level (developed DSP in figure 4.12), is presented in figure 4.13. Each of the individual blocks is explained in the following subsections. The top level receives 16 bit signed samples at 976.56 kHz, which means, it receives one sample every 128 clock cycles.

As depicted in the figure, the received samples start by being filtered by a low-pass, this filter is used as anti-aliasing. The samples outputted by the low-pass filter are then decimated by a factor of four and are filtered by a band-pass filter, this band-pass filters around the frequencies used by the system. After the filtering process is complete, the samples are stored and organised in RAM's. These RAM's hold at all times, the last 10200 samples necessary for the correlation (10200 is the number of samples corresponding to a full signal). From the "Samples organiser" module, the samples are received by three correlators, two of them are used to compensate for Doppler. Each of the results of the correlators is received by a module that identifies a correlation peak. This module then outputs the results to the AXI. The AXI makes the interface with the ARM processor where the results are acquired.

In the figure, the module named "Correlation Buffer" holds the results of the correlation that surround the correlation peak. This was implemented so it would be possible to observe the shape of the correlation peak at every result.

This top-level module outputs the value of the correlation peak, the timestamp of the correlation peak, the index of received sequence and the 128 results of the correlation that surround the correlation peak.

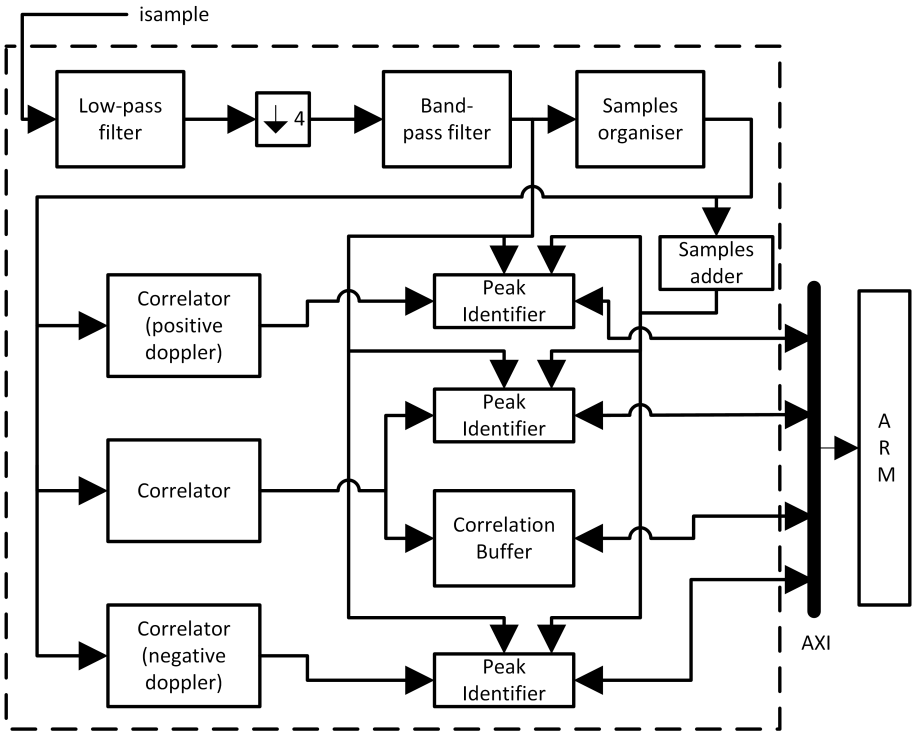



Figure 4.13: DSP Top Level Block Diagram

#### 4.2.2.3 Filters

This subsection is relative to the low-pass and band-pass blocks depicted in the top level. The sampling frequency of the analogue to digital converters is 125MHz. As previously shown, the wrapper used for the developed system, integrates a decimator. This decimator can be configured to decimate by any power of two.

Figure 4.14 shows the frequency response of the low-pass filter used in the implementation of the system. This filter was used as anti-aliasing before the final decimation. The analogue front-end filters have a cut-off frequency of approximately 250 kHz. This cut-off frequency means that the Nyquist frequency is 500 kHz. This is the sampling frequency below which aliasing may occur.

Due to the analogue filters Nyquist frequency, 128 is the highest power of two that can be used as a decimation factor without risking incurring in aliasing.

$$\frac{125000000}{128} = 976.56\,\text{kHz} \tag{4.1}$$

This low-pass filter is used so the samples can be further decimated.

The −3 dB pass-frequency of the filter is 35 kHz and the −200 dB cut-off frequency is 120 kHz. The filter was designed to this specifications, however, the implementation was done using fixed point. To do that, all the coefficients of the filter were multiplied by $2^{10}$ and rounded up. Once the filtering process is through, the result is shifted 10 bits to the right, so the gain of the filter stays 0 dB. This process causes the filter response to change. Figure 4.14 shows the filter response after the approximations previously stated. The cut-off band is no longer attenuated at −200 dB.

The filters order is 127, which means, it has 128 coefficients. This number was chosen taking into consideration that 128 clocks are available between two consecutive samples, making it possible to perform all necessary calculations sequentially, using only one multiplier and one adder.

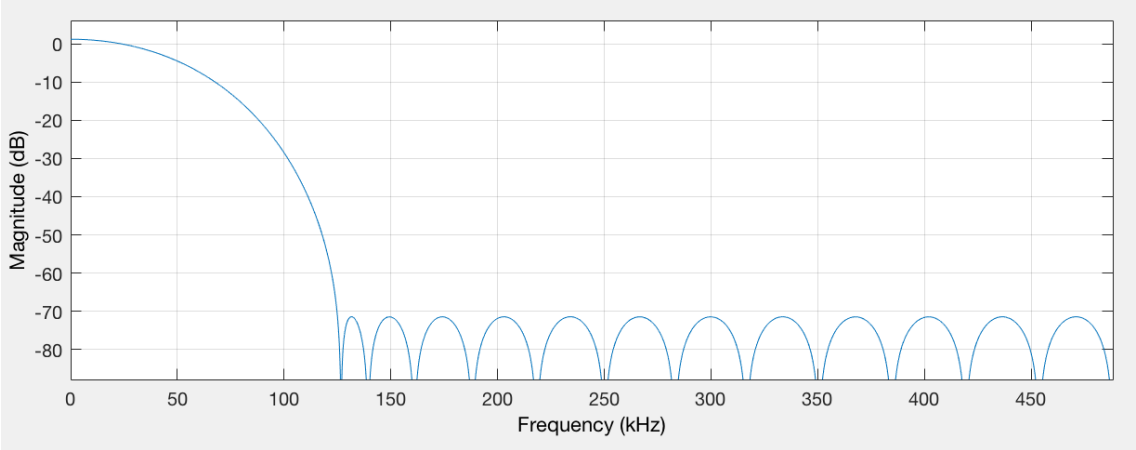



Figure 4.14: Low-pass frequency response

After the low-pass filter is applied, the samples are decimated again by a factor of 4. This new decimation makes it a total decimation of 512 from the initial sampling frequency, only possible without incurring in aliasing after the low-pass is applied.

Again, the order of this filter is chosen taking into account that 512 clocks are available between two consecutive samples, and so, the filter has 512 coefficients, making it a filter with an order of 511.

This filter is used to attenuate possible noise outside the used frequencies. In the low frequency side of the band-pass, the −3 dB pass-frequency is 18 kHz and the −80 dB cut-off frequency is 16.5 kHz. In the high frequency side, the −3 dB pass-frequency is 31 kHz and the −80 dB cut-off frequency is 32.5 kHz. Figure 4.15 shows the frequency response of the designed filter, the effective response is not to the designed specifications for the same reasons previously stated for the low-pass.

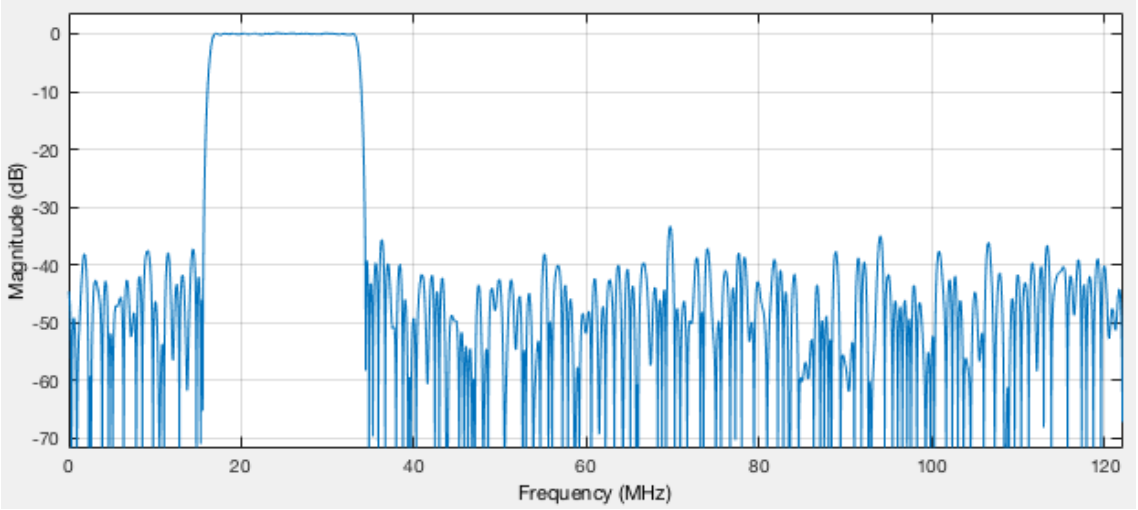



Figure 4.15: Band-pass frequency response

Figure 4.16 roughly illustrates the digital implementation of the low-pass filter. It receives a signed 16-bit sample every 128 clocks. Always keeps in a buffer the last 127 samples, to do that, it is used a 16 by 127 RAM. A ROM of 16 bits by 128 addresses holds the coefficients of the filter.

When a new sample is received, the calculations for the first 127 coefficients are already preformed, this way, the module only introduces a two clock delay. The figure depicts three levels of registers. However, the accumulator register is bypassed when a new sample arrives.

This implementation uses the minimum amount of resources considering that all the processing is done sequentially.

The multiplication result register is introduced solely for timing purposes.

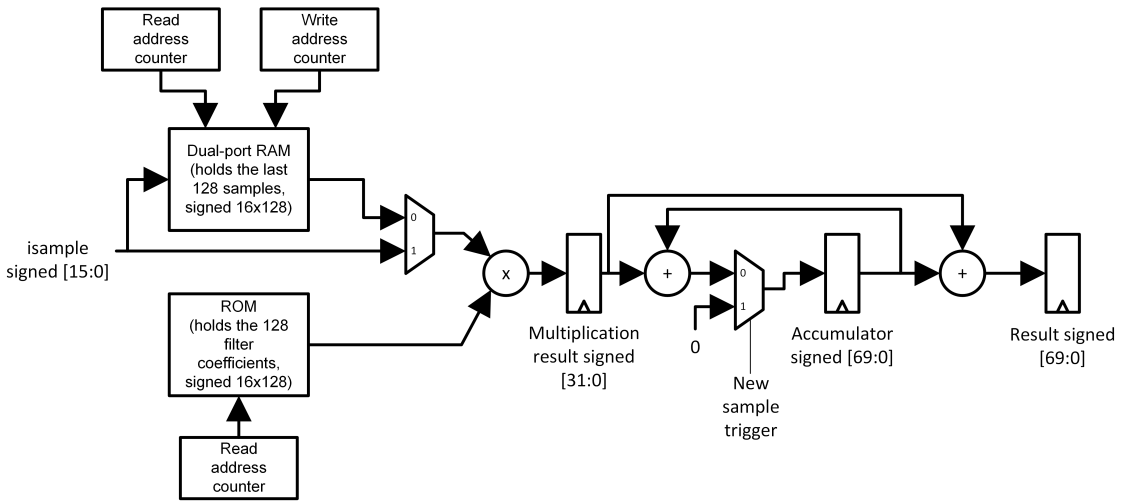



Figure 4.16: Filter digital implementation

The band-pass filter is implemented by the same logic, with the only difference that the RAM and ROM have 512 addresses instead of 128.

#### 4.2.2.4 Samples organiser

A full correlation is performed between two consecutive samples, two consecutive samples are separated by 512 clock cycles, and a complete correlation processes 10200 samples. This means that at least $10200/512 \approx 20$ samples have to be processed per clock cycle. To achieve this, the last 10200 acquired samples must be organised in such a way, that 20 consecutive samples are available in parallel per clock. This is implemented using 20 dual-port RAM blocks through which the incoming filtered samples are distributed. The first sample is stored in RAM 0, the second in RAM 1 and so on. This way, it is always possible to have 20 consecutive samples in parallel. Figure 4.17 demonstrates how the module is implemented.

Each RAM has a controller block. Each controller activates the write enable when the received sample is to be stored in the RAM it controls. Every time a new sample arrives, that sample substitutes the oldest stored sample. Once a correlation cycle starts, the RAM controllers output the stored samples from the oldest to the newest.

This implementation poses a problem to the module that receives the 20 samples because the order of the outputted samples changes every time a new sample arrives. For example, after a sample is written in RAM 0, the temporal order of the 20 outputted samples is 1,2...18,19,0.

It is not viable to use multiplexers to keep a steady order. Twenty multiplexers of 20 to 1 would be needed, that would be too resource consuming, considering that each sample has 16 bits.

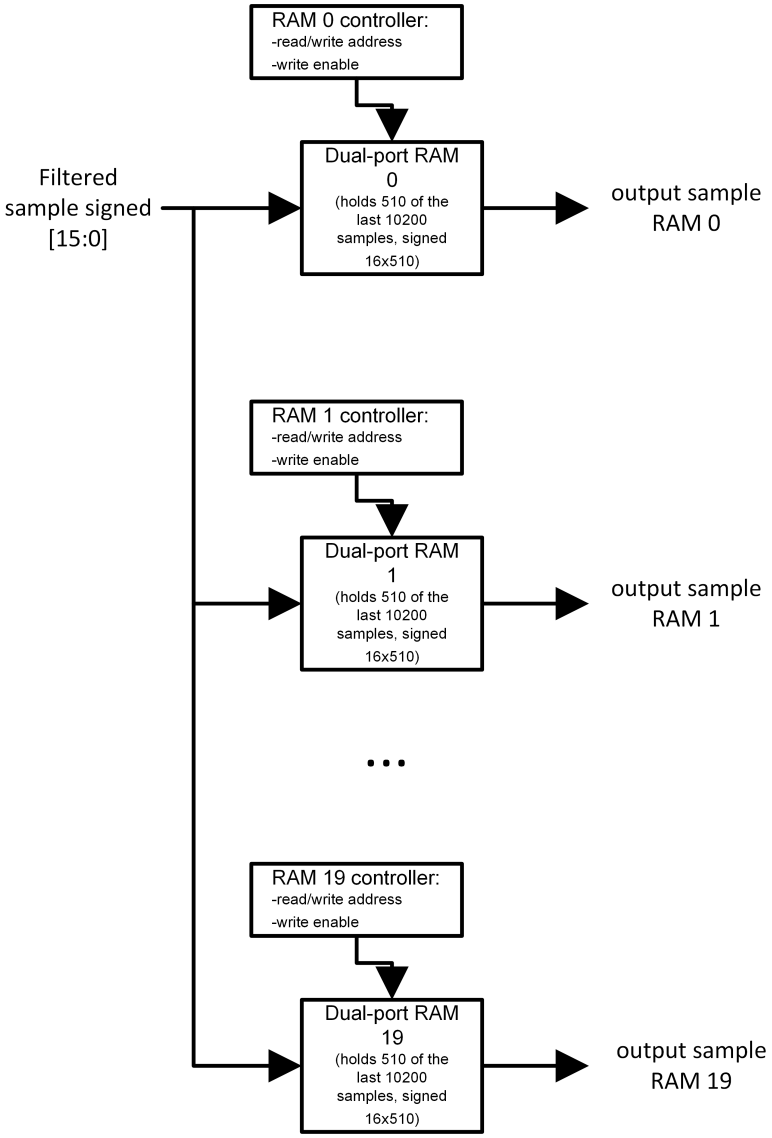



Figure 4.17: Samples organizer diagram

#### 4.2.2.5 Correlator

The correlator module receives as input the 20 samples from the samples organiser module. The 20 samples represent 2 periods of the carrier frequency. As explained in 3.5, the correlation consists in adding or subtracting the sample to the correlation accumulator. If the samples in the inputs were always ordered from 0 to 19, that would mean, adding samples from 0 through 4, 10 through 14 and subtract the others. However, as explained in the previous subsection, the order of the samples from the samples organiser changes. Individual correlation units are used to overcome this problem. Each of the correlation units receives two of the 20 samples, receives sample #x and #(x + 10). Ten samples represent one period of the fundamental frequency, so, the same action is to be performed on sample #x and sample #(x + 10). These units, green boxes in figure 4.18, keep

track of the samples temporal order and either output the value of the samples, if they are meant to be added to the correlation accumulator, or their symmetric value otherwise.

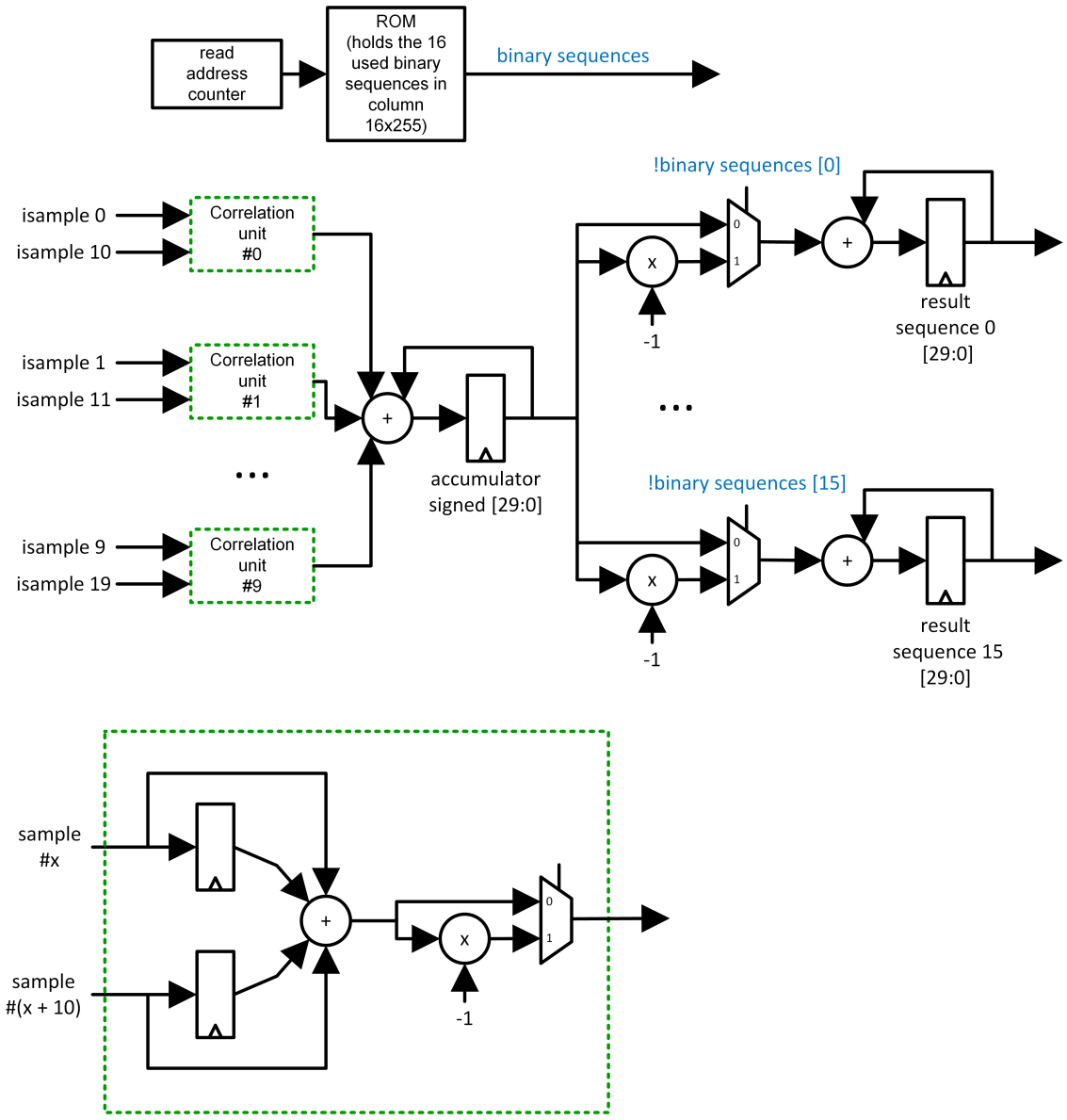



Figure 4.18: Correlation module diagram

Each bit of the pseudo-random binary sequence modulates 4 periods of the fundamental frequency, because of this, the correlation unit module doesn't output a result every clock. First, it receives 2 of the 20 samples, in the next clock, receives 2 of the next 20 samples, and only then, does it output the result of the four samples. This result concerns one bit of the binary sequence.

The results of all correlation units are then added every two clocks. This is the result of the correlation of 4 periods, assuming that the 4 periods were modulated by the binary value 1.

Also represented in figure 4.18 is a ROM, this ROM holds the 16 pseudo-random binary sequences used to modulate the signals. The sequences are stored by column, this way, every address

of the ROM holds a bit of each of the 16 different sequences. When the correlation process starts, the ROM starts being read from the address 0. The read address is then incremented every two clocks.

As depicted in figure 4.18, the result of the sum of the correlation units, is then added or subtracted to each of the final 16 accumulators. These 16 accumulators hold the result of the correlation for each of the binary sequences. Whether the result of the sum of the correlation units is added or subtracted, depends on the bit stored in the ROM for each of the 16 sequences.

On the limit, the correlation matches perfectly, and every received sample occupies the full range of the 16 bits in terms of magnitude, that equates to a maximum possible correlation magnitude of:

$$(2^{15} - 1) \times 10200 \approx 3.3 * 10^8 \tag{4.2}$$

This value can be held by a 30 bit wide accumulator without risking overflow. This is the with used for the accumulators.

#### 4.2.2.6 Correlation peak identifier

This module receives the 16 correlation results, one per binary sequence. It identifies a correlation peak and the received sequence.

The module has a configurable threshold above which it assumes a signal is being received. When that threshold is reached, the module starts searching for a correlation peak within a time frame. The threshold is compared against the result of the band-pass filter. When the result of the band-pass filter is higher than the threshold, the module starts storing the maximum correlation value received from each of the 16 correlations, updating that maximum every time an higher result occurs. When that higher result is stored, along with it, is stored the timestamp. The search for the peak is performed by default during 20000 samples after the threshold is reached, this is a configurable parameter. This number was defined based on laboratory experiments, it is enough time for the reverberations to dissipate. After the search window is complete, it is identified the highest correlation peak out of the 16.

The band-pass filter is steep around the used frequencies. However, noise at a frequency between the 18 and the 31 kHz can trigger the module to start searching for a correlation peak when no signals from the system are being received. This will yield a random time of arrival. Also, it is not sufficient to look at the magnitude of the correlation peak to establish whether it is a real peak or not, for its magnitude depends on the intensity of the received signal. It was necessary to implement a peak validation method.

If a signal is received free of any noise and distortion, the correlation peak has a magnitude equal to the sum of the absolute value of the 10200 samples that represent the signal. In real conditions, when noise and distortion are added, the correlation peak will no longer match exactly that value. But it will be within a significant percentage.

Figure 4.19 shows the results of a field test where there was noise in the used frequencies. That noise had the same intensity of the signals of the system. The top plot of the figure shows the received recorded signal after the filtering process. The 3 zones where the signal is slightly more intense, correspond to the moments when a signal of the system was being received.

The second plot of the figure shows the spectrum. This spectrum was calculated in a time gap of the recorded signal where no signals of the system were being received. Therefore, this is the spectrum of the noise.

The third plot depicts two graphs, the one in orange is the result of the correlation, the one in blue is the result of adding the absolute values of 10200 samples in a sliding window. The bottom plot is the ratio between the graphs shown in the third plot. It is evident that the ratio between the two is significantly higher when a valid correlation peak occurs. This property is used in the system to validate a presumed correlation peak.

If a peak is validated, this module outputs the magnitude of the peak, the timestamp and the information concerning which sequence was received. If the peak is invalidated the search window is extended.

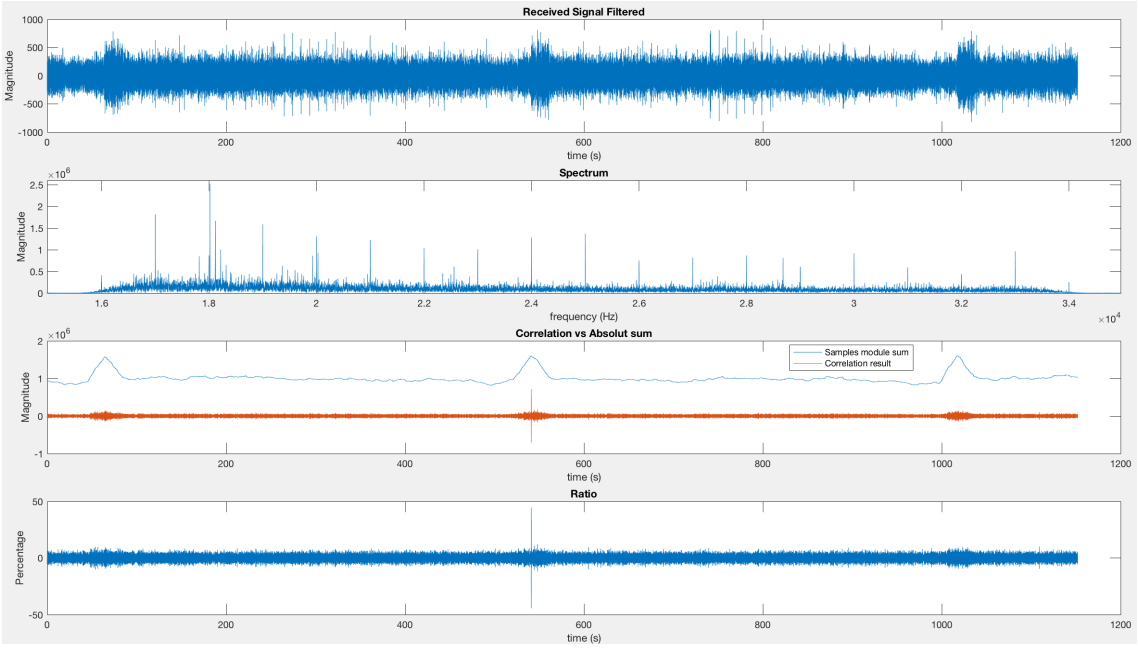



Figure 4.19: Correlation result obtained in the presence of noise

#### 4.2.2.7 Correlation result buffer

This module was implemented so it would be possible to observe the correlation peak in real time. The module stores 128 correlation results that surround the correlation maximum for each of the 16 correlations. Once a correlation peak and the corresponding sequence has been identified, the module outputs the stored 128 correlation results corresponding to that sequence.

The module stores the 16 correlation results as they are received. Each correlation result is stored by one of two buffers of 128 addresses. One of the buffers stores the values as the results are received. The other buffer holds the results of the correlation that surrounds the previous maximum. When a new maximum occurs, the buffers switch functions. The module does this, up until the peak identifier, signals that a correlation peak corresponding to a time of arrival has identified. Then the module outputs the stored 128 results corresponding the received sequence.

The module needs to store $128 \times 2 \times 16 = 4096$ correlation results of 32 bits each. The used FPGA has configurable dual-port RAM's that can be used in blocks of 1024 lines of 32 bits. Four of these block RAM's were used so other resources would be saved, (registers and LUT's). Each RAM was organised as depicted in figure 4.20. Each of the 16 sequences takes up 256 addresses. The figure represents roughly how the module is implemented, only one of the four RAM's is depicted.

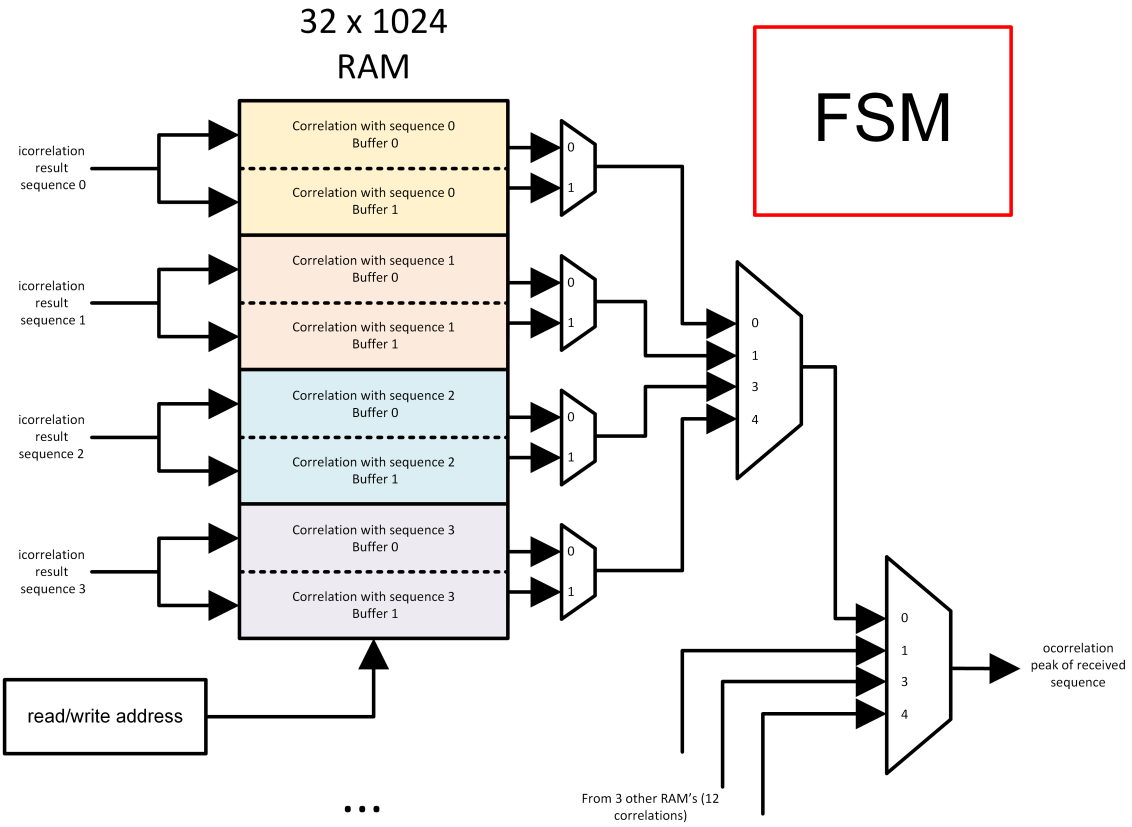



Figure 4.20: Buffer of correlation results

#### 4.2.2.8 Correlator with Doppler compensation

This module implements the Doppler compensation method presented in 3.9. The sub-module presented ahead is integrated between the samples organiser and a correlator. As previously ex-

plained, the method consists in removing samples to compensate negative Doppler and add samples to compensate positive Doppler.

This module consists of 20 buffer registers between the samples organiser and a correlator. Each of its outputs can come from the buffers or directly from the samples organiser.

This module has a configurable parameter that determines the number of received samples between an addition or a removal of a sample. It generates an internal trigger to add and remove samples.

To compensate for positive Doppler, the module adds samples periodically. Which means, it periodically delays the samples that are received from the samples organiser. The module initially outputs all the samples directly from the inputs. When a trigger signal occurs, it buffers the oldest sample in the input. In the next clock, instead of outputting directly the sample from the input, it outputs the sample in the buffer. Next time a trigger is received, does the same thing to the now oldest sample, and so one. The frequency of the trigger determines the frequency at which samples are added.

To compensate for negative Doppler, the module needs to remove samples. To do this, the module does the opposite function. It starts by buffering all the inputs, and every time a trigger occurs starts outputting one of the samples directly from the input. In this case, it does it from the newest to the oldest sample in the inputs.

This module precedes a correlator. As explained in 4.2.2.5, a correlator as internal counters that keep track of the order of the 20 samples in the inputs. Every time a sample is added or removed, the temporal order of the correlator inputs changes. Because of this, the correlation units also receive the trigger to add or remove samples so that they can update the order of the samples in the internal counters.

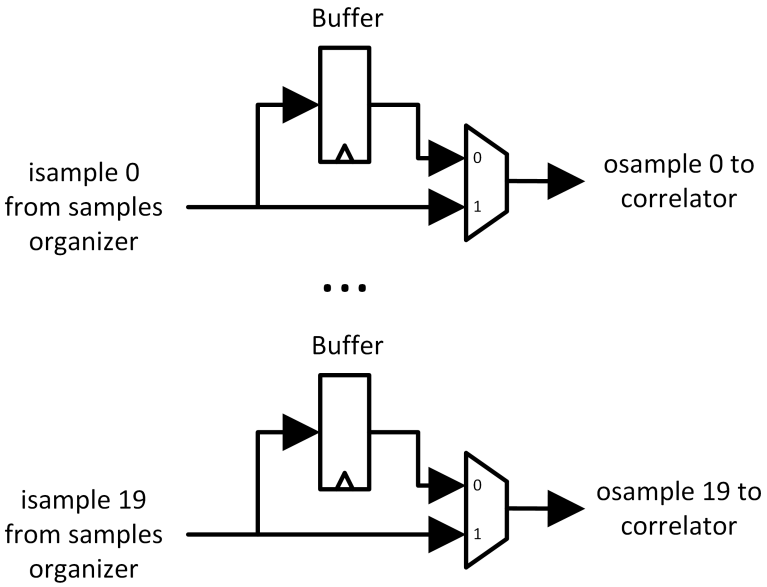



Figure 4.21: Buffers necessary to compensate Doppler

#### 4.2.2.9 Adjustable parameters of system

The following parameters are configured by the ARM processor when the system is loaded:

- Correlation peak search window size - As explained, when the signal from the band-pass filter reaches a certain threshold, the system starts searching for a correlation peak. This parameter is relative to the number of correlation results that the system searches through. Default: 20000 results;

- Threshold - When the results of the band-pass filter reach this threshold, the system starts searching for the correlation peak. Default: 800 (2.44 % of the full range);

- Doppler compensation - This parameter sets the rate at which samples are added and removed by the Doppler compensation modules. Default: 1200 samples between addition and removal of samples.

#### 4.2.2.10 Implementation results

Table 4.1 shows the FPGA resource occupation post place and route without the two correlators that compensate for Doppler. This implementation included the wrapper of the system.

| Resource | Utilization % |
|---|---|
| LUT | 51 |
| FF | 18 |
| BRAM | 49 |
| DSP | 3 |

Table 4.1: Implementation results without Doppler compensation

Table 4.2 shows the FPGA resource occupation post place and route after one of the two correlators that compensate Doppler are added. The addition of this module increases the percentage of used LUT's from 51% to 81%. However, the post syntheses occupation of one correlator with Doppler compensation is 13% of LUT's, as shown in table 4.3. There was no time to understand this increment in resources utilization. But this way it is not viable two have the three correlators working in parallel.

| Resource | Utilization % |
|---|---|
| LUT | 81 |
| FF | 27 |
| BRAM | 50 |
| DSP | 3 |

Table 4.2: Place and route results with main correlator and one that compensates Doppler

| Resource | Utilization % |
|---|---|
| LUT | 13 |
| FF | 6 |
| BRAM | 1 |
| DSP | 0 |

Table 4.3: Correlator that compensates for Doppler

## 4.3 Software processing

The ARM processor runs a developed software that initializes the system and sets all the parameters. The software starts by reading a configuration file, this file contains all the configurable parameters. To communicate with the FPGA, the software opens a mounted character device and instantiates a memory map that corresponds to the addresses used in the FPGA. Once the system is running, the software reads the address that corresponds to a trigger signaling that a time of arrival has been estimated, this trigger is acquired by polling. After the trigger, the software reads the magnitude of the correlation peak, the timestamp, the number of the received sequence and the 128 results of the correlation that surround the correlation peak.

After the previously stated results are acquired, the software also applies the algorithm presented in 3.7 to the 128 correlation results. The 128 results and the result of the applied algorithm are then transmitted via an UDP socket. The socket is read by script running in Octave that plots both the correlation result and the result of the applied algorithm. This was implemented so it would be possible to observe the shape of the correlation peak in real time.

All the information received from the FPGA and the results of the applied algorithm are stored in a file.

# Chapter 5

# Results

This chapter presents and analyses the real-time results of the implemented system. The first tests were performed in a laboratory environment. After the laboratory results were validated, the system was tested on the field.

## 5.1 Laboratory tests

The first tests were carried out in a $4 \times 4 \times 1.6$ meter deep laboratory test pool, filled with fresh water. In these conditions, the underwater sound speed is around 1481 m/s, that was the assumed sound speed during the test.

The receiver needs to know when a signal is transmitted to be able to count the time lapse between transmission and arrival. With that information, the receiver can then calculate its distance to the transmitter based on the sound speed. The accuracy of the time of arrival estimation is judged based on the difference between the estimated distance and the real distance.

In the laboratory tests, the transmitter and the receiver were wired together to synchronize the two. The receiver sends a trigger requesting a signal transmission and then counts the time between the trigger and the time of arrival estimation.

### 5.1.1 Setup and parameters

The parameters of the test were:

- Signal detection threshold - 800 (2.44 % of the full range);
- Correlation peak search window - 20000 samples;
- Gain of the two I2C programmable amplifiers, (Analogue front-end) - 5;
- Signals per second - 8, maximum rate without interference between signals.

For all tests, the transducer was placed at one end of the pool. The hydrophone was attached to a cable on two pulleys that could be dislocated across the pool. This cable moved in line with the transducer making it easy to test different distances between transmitter and receiver.

### 5.1.2 Results

The tests consisted in placing the hydrophone in several positions equally spaced by 15 cm. Five thousand results were obtained for each position. Figure 5.1 shows the results of the tests, each colour is the results obtained for one position. This image shows the results of eleven different positions of the hydrophone. The results corresponding to one position will be examined ahead in more detail. For now, notice that the legends on the figure show the results separated by 15 cm.

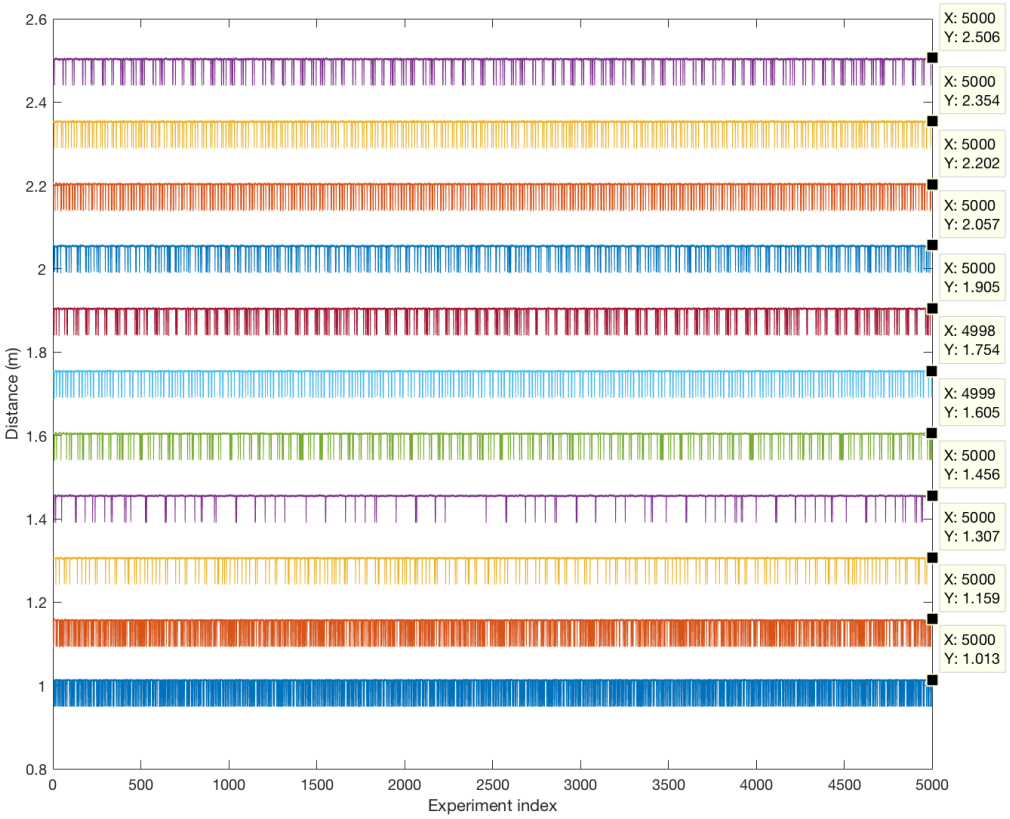



Figure 5.1: Estimated distances for equally spaced positions

The plot in figure 5.2 shows the error between the real distance and the average of the results for each distance. The error of the estimated distances is always equal to, or less than 5 mm.

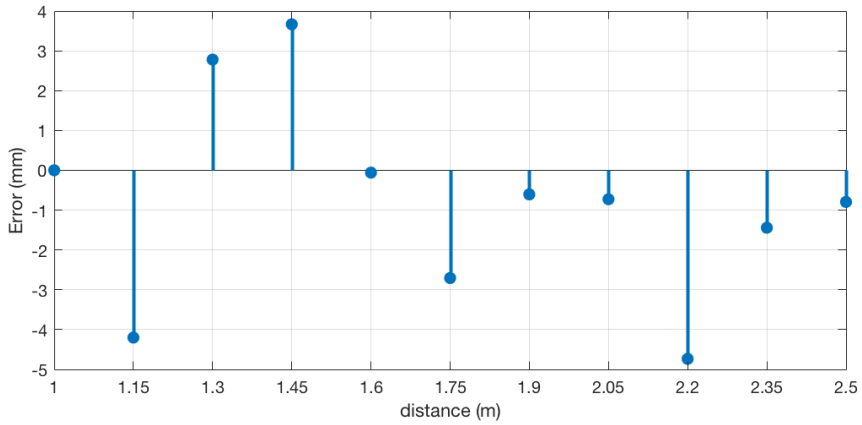



Figure 5.2: Real distance versus estimated distance

Figure 5.3 shows the results obtained with the transmitter and receiver separated by 1 m. It is possible to see that all the results are concentrated either in the upper or on the lower part of the plot, being the upper part the denser of the two. The red and the yellow lines in the figure represent the average of the points surrounding those two areas. The two legends show that the two averages are separated by 6.17 cm. This results from the problem presented in 3.7. As explained, due to distortions such as reflections caused by multipath, the maximum of the correlation may not represent the instant of arrival. The correlation peak is formed by several local peaks separated by 10 samples. If one of the local side peaks has an higher magnitude than the one that represents the time of arrival, the error concerning distance is approximately 6 cm, which explains the difference between the upper and the lower results.

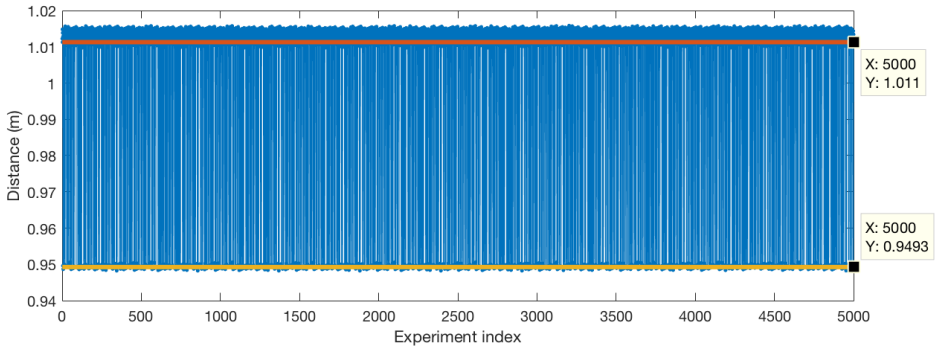



Figure 5.3: Results for a fixed position

Figure 5.4 shows an histogram of the results presented in figure 5.3. This figure shows the distribution of the results yielded. As explained, the results form two well-defined distributions separated by approximately 6 cm.

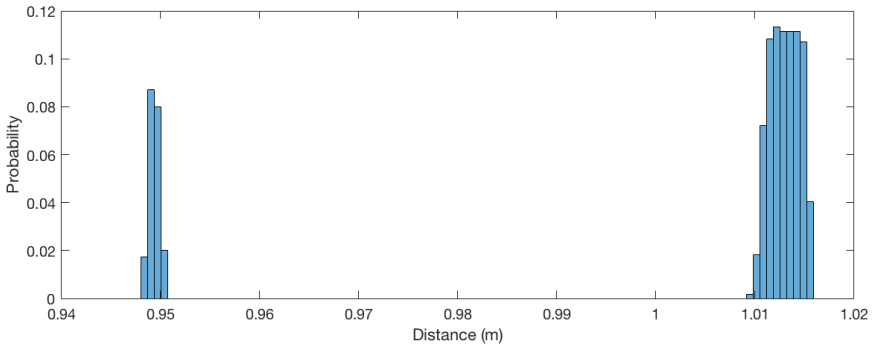



Figure 5.4: Histogram of the results for a fixed position

Figure 5.5 shows the same result as in figure 5.3 represented in blue. On top of that, represented in orange, is shown the results after the algorithm presented in 3.7 to improve the correlation peak has been applied. The red and the green lines are the average of the upper and lower results for each of the two plots. It is clear that after the correlation peak improvement, the difference between the upper line of the results and the lower line is significantly smaller. Previous to the improvement, the two distributions were separated by 6.17 cm, after the algorithm is applied, the difference reduces to 4.1 cm, the standard deviation reduces from 0.0257 m to 0.0168 m. This shows that the method to improve correlation peak considerably increases the precision of the system.

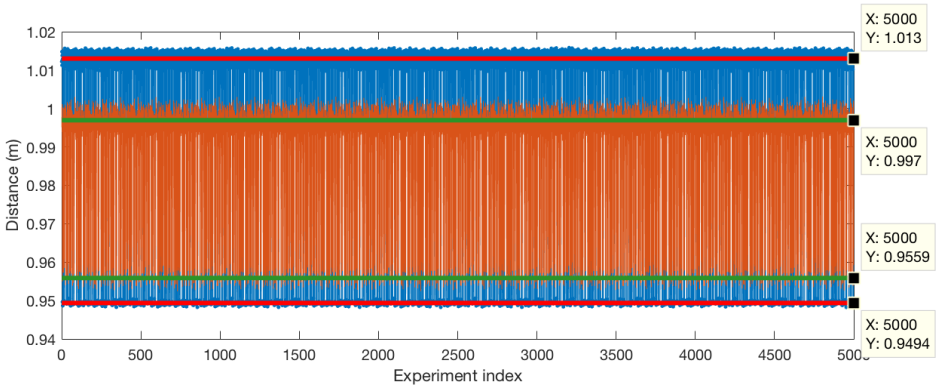



Figure 5.5: Estimated position, before versus after peak adjustment

The correlation peak improvement algorithm is a useful method to increase precision if a high frequency of distance measurements is necessary. Figure 5.6 presents the results of another approach to improve precision. In this method, measurements are grouped in groups of 8 consecutive results, the upper and lower lines of the 8 results are identified, and the final result is the average of the samples situated in the line with more occurrences. This solution reduces eight times the number of measurements per second. However, the precision of the system increases radically. After this algorithm is applied, the results are within a range of 2 mm. The standard deviation in

this case is 0.34 mm. However, this method can only be applied if the receiver is stationary during the 8 measurements.

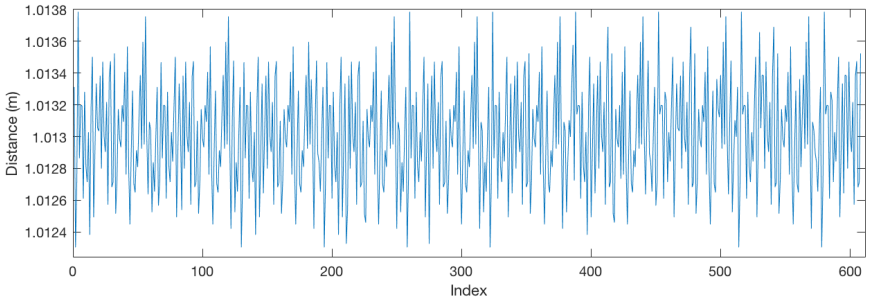



Figure 5.6: Distance estimation after processing results in groups of 8

Figure 5.7 shows the results for one of the 16 signals used by the system. There is a clear pattern in the several results across time. This pattern is most likely caused by the clock drift between the FPGA used for the transmission and the FPGA used in the receiver. This clock drift also contributes to the oscillation in the results.

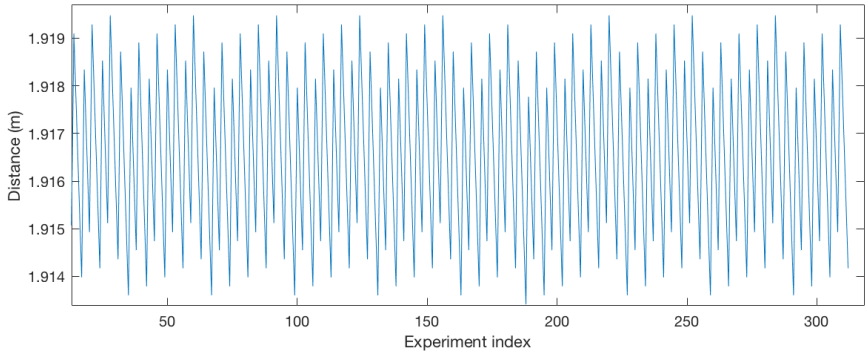
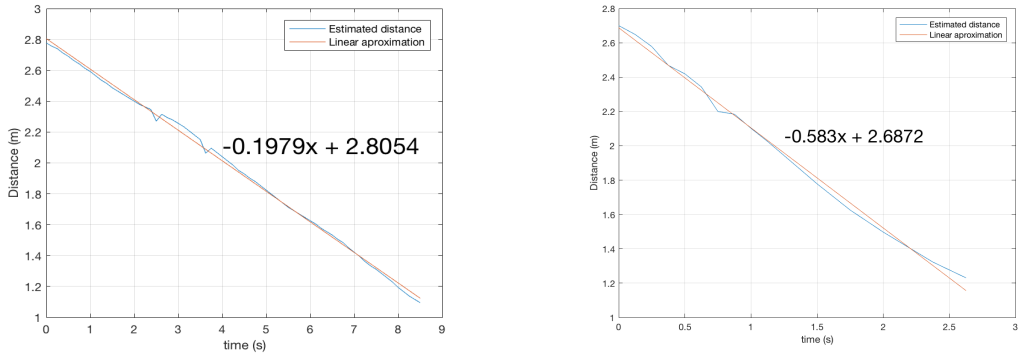



Figure 5.7: Estimated distance using only one of the 16 possible signals

The two figures in 5.8 show the results acquired with the receiver moving relative to the transmitter. In the two tests, the hydrophone was manually pulled at close to a constant speed. The linear approximations of the results show that the hydrophone was moving roughly at 0.1979 m/s in 5.8a and moving at 0.583 m/s in 5.8b. The actual speed at which the hydrophone was moving is unknown because it was manually pulled, but the fact that the results follow a linear variation shows that the system is robust to relative movement between transmitter and receiver.




(a) Moving at 0.1979 m/s (b) Moving at 0.58 m/s

Figure 5.8: Estimated distances with the receiver moving

The tests were preformed with the 16 binary sequences being transmitted in round-robin. In the 55000 acquired results the sequences were identified without errors. Figure 5.9 shows the index of the received sequence for a portion of the experiment.

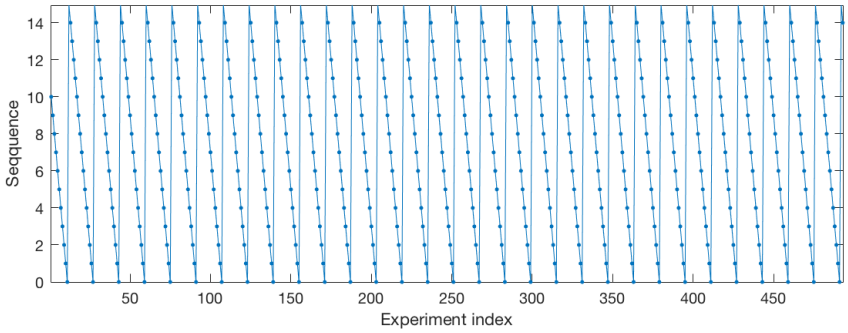



Figure 5.9: Sequences received across time

## 5.2 Field test

After the system showed good results in the laboratory tests, the next step was to test the system on the field. The placements of the transmitter and receiver are marked in figure 5.10. The transducer was placed on the left side of the image, where all the arrows meet. The hydrophone was placed at three different distances from the the transmitter, 11 m, 118 m and 147 m.

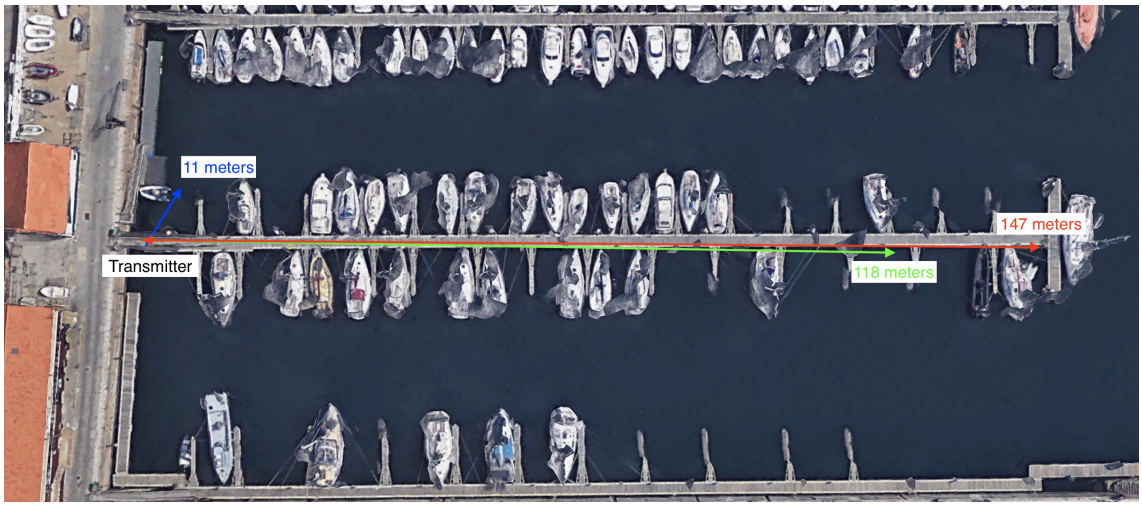



Figure 5.10: Field test positions

#### 5.2.0.1 Setup and parameters

The distance estimations presented in the following results were calculated considering a sound speed of 1500 m/s. This was not the exact sound speed of the environment at the time, but it is a common value to use. The most important thing to understand with these tests is how precise the system is. The precision of the system is accessed by analyzing the standard deviation of the results for each position. The precision is not affected by whether the distance is exact or not. Therefore, it is not crucial to use the exact sound speed in the calculations.

There was noise in the used frequencies during the test, and at the time, the peak validation method presented in 4.2.2.6, was not yet implemented. The trigger for the system to start searching for a correlation peak was based on a threshold after the band-pass filter. The noise, triggered the system to search for correlation peaks when no signals of the system where being received. That resulted in obtaining random time of arrival results. These random results are identifiable and were removed from the results presented ahead.

The Pulse Per Second (PPS) signal of two GPS's were used to synchronise the transmitter and the receiver. Every time a PPS was received, the transmitter would emit a signal, and the receiver would start counting the number of clocks cycles until the time of arrival.

The parameters of the test were:

- Signal detection threshold - 800 (2.44 % of the full range);
- Correlation peak search window - 20000 samples;
- Analog front-end gain - Dependent on distance.

### 5.2.1 Results of the first test (11 meter distance)

A probability histogram was created to isolate the random time of arrival estimations resulting from noise. Figure 5.11 shows the created histogram. As represented, 98 % of the results are situated between 10.4 m and 12.15 m. Because this range concentrates so many results, it is unlikely that they are random. Therefore, only results within this range are considered.

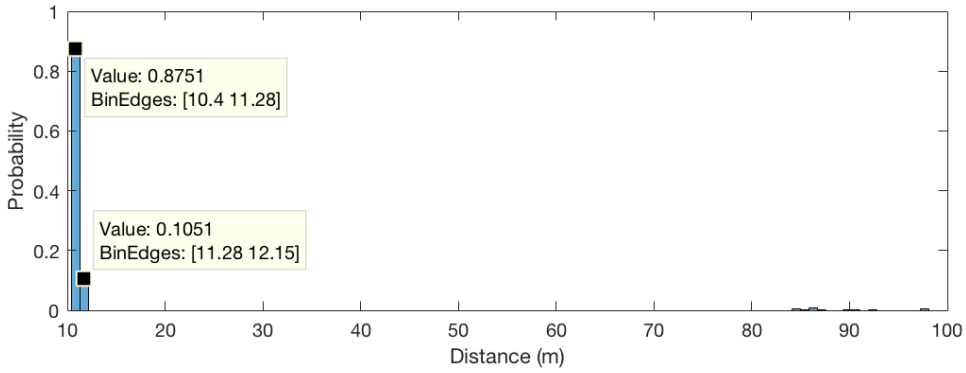



Figure 5.11: Histogram of the results for a distance of approximately 11 meters

In this test, the hydrophone was attached to a boat which was not a fixed point. It was floating in the water within a confined area. Figure 5.12 shows the results obtained in this test. Each dot of the plot is a time of arrival estimation converted to distance. The estimated distance results range from 11.04 m to 11.30 m. No conclusions can be derived from these results concerning the precision of the system because the hydrophone did not have a fixed position.

The orange line in the plot represents the average of the results. As previously stated, the boat where the hydrophone was attached was slightly moving, but this average distance matches the measured distance.

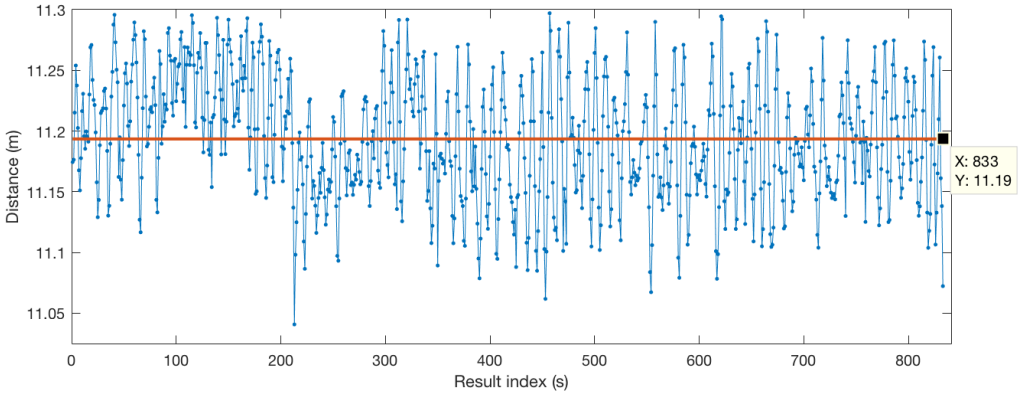



Figure 5.12: Distance results when approximately 11 meters away form the transmitter

### 5.2.2 Results of the second test (118 meter distance)

During this test, the hydrophone was in a fixed position relative to the transmitter. Figure 5.13 shows the histogram of the obtained results. The random results follow a practically uniform distribution. Only results ranging from 117 m to 119 m are considered in the results presented ahead.

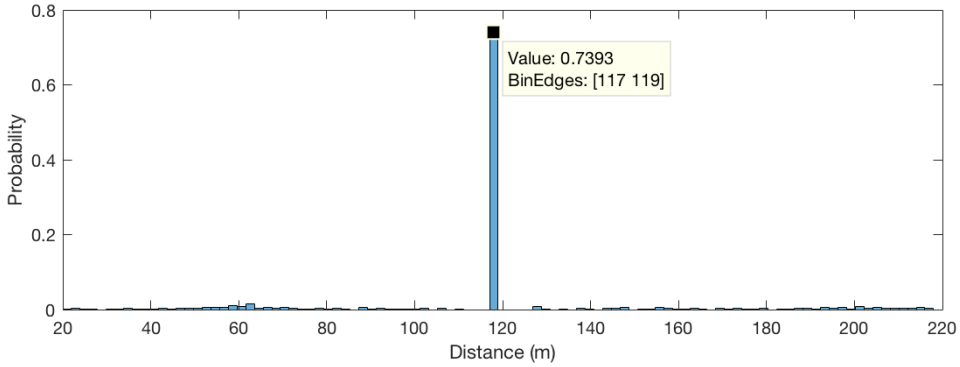



Figure 5.13: Histogram of the results for a distance of approximately 118 meters

Figure 5.14 shows the results after the random values have been excluded. The presented plot shows that most of the results concentrate around two levels.

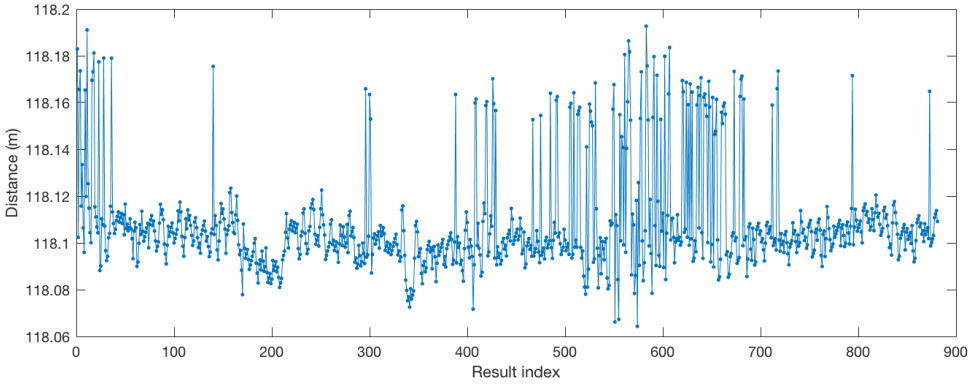



Figure 5.14: Distance results when 118 meters away form the transmitter

Figure 5.15 shows a histogram of the results within the accepted range. Two distinct distributions form separated by 6 cm. As previously stated, this is the separation between two consecutive local maximums that form the correlation peak.

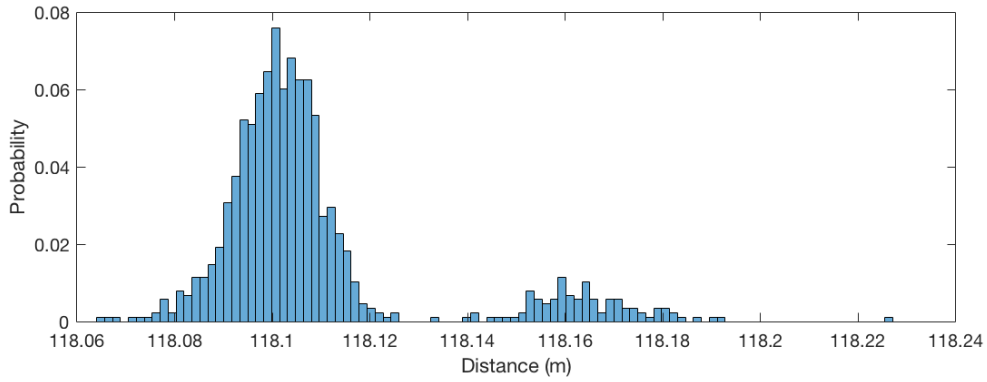



Figure 5.15: Histogram of the accepted results for a distance of approximately 118 meters

Figure 5.16 compares the results before and after the peak adjustment algorithm is applied. The blue plot is the before, the plot in orange is the after. As it happened in the laboratory tests, the precision of the system increases. Prior to adjustment, the standard deviation of the results was 2.18 cm, after the correlation peak improvement the standard deviation decreased to 1.69 cm.

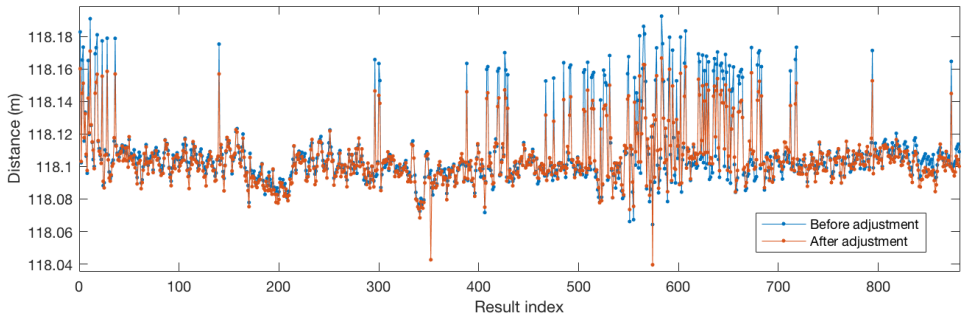



Figure 5.16: Distance results when approximately 118 meters away form the transmitter before versus after correlation peak adjustment

### 5.2.3 Results of the third test (147 meter distance)

At the distance of this test, the received signals from the transmitter and the noise had approximately the same magnitude. As shown in 5.17, that produced more random incorrect results. Only approximately 57 % of the results are a consequence of receiving a signal from the transmitter of the system.

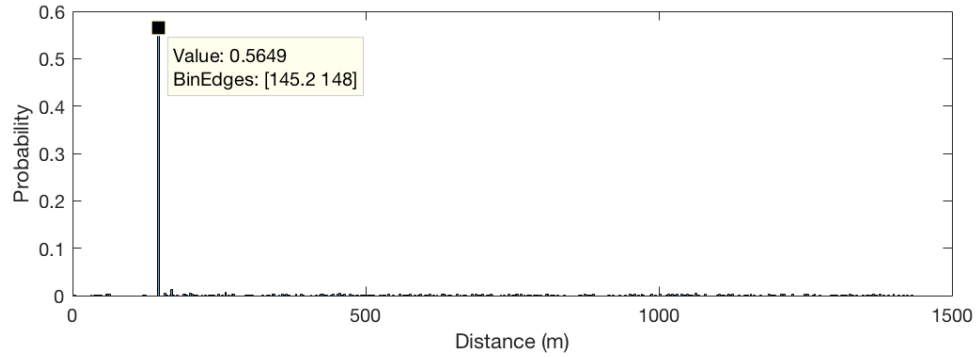



Figure 5.17: Histogram of all acquired results for a distance of approximately 147 meters

Figure 5.18 shows the estimated distances after the random results are excluded.

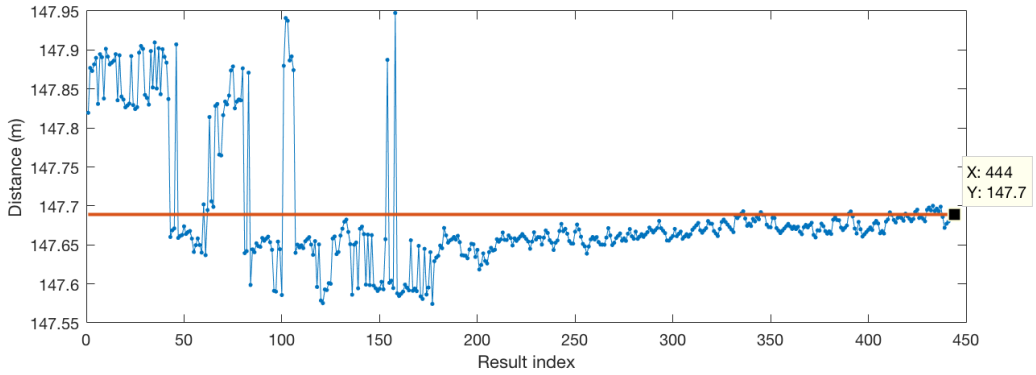



Figure 5.18: Distance results when approximately 147 meters away form the transmitter

Figure 5.19 shows an histogram of the results from the accepted range. The results form 4 distributions, two on the right of the histogram and two on the left, in both cases separated by 6 cm. These two groups of distributions formed probably due to multipath.

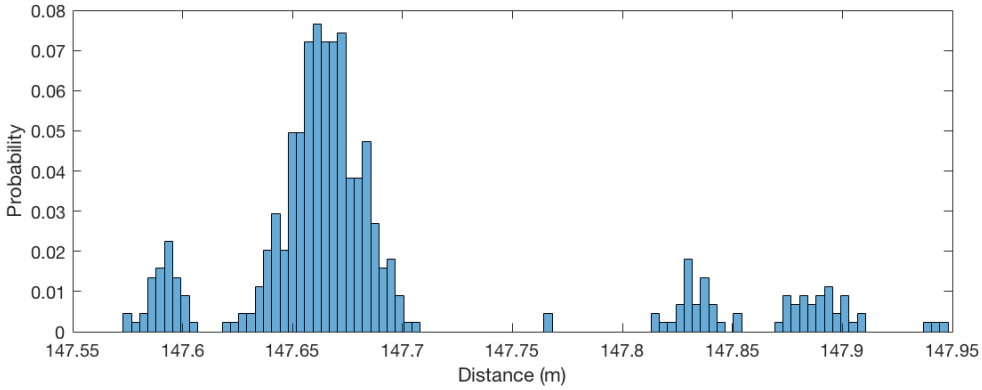



Figure 5.19: Histogram of the accepted results for a distance of approximately 147 meters

## 5.3 Results Discussion

The worst result obtained in the test tank was when the receiver and transmitter were separated by one metter, in this setup the standard deviation of the obtained results was 1.68 cm, the standard deviation of 118 meters was 1.69 cm. This shows that the system is capable of maintaining a good precision as the distance between transmitter and receiver increases.

As stated in the introduction, this work was intended to replace the system currently used in the AUV's developed by INESC TEC. The exact numbers relative to the precision of the system currently in use are not known, but it is acceptable to state that the results here presented show that the developed system is considerably more precise.

The developed system also presented good results when compared to the systems presented in chapter 2.

# Chapter 6

# Conclusions

The objective of this dissertation was to improve the time of arrival estimation of underwater acoustic signals.

Chapter 5 presents the obtained results, both in the laboratory test tank and in the field. The presented results show that the system has a good precision and that the precision does not decrease significantly with the increment of the distance between transmitter and receiver. However, more tests should be carried out to understand the limits of the system.

The algorithm proposed in chapter 4 to improve the correlation peak, considerably increased the precision of the system. The proposed method to compensate for the Doppler effect was implemented, however, due to lack of time it was never tested. One possible way to accurately test if the implementation works is by changing the clock frequency of the transmitter. That way, it is possible to induce a controlled frequency deviation and discover precisely how much Doppler effect the system can tolerate.

The established communication channel is a channel of low bit rate. However, it is reliable, in all performed tests no communication errors were identified. The highest tested bit rate was 32 bit/s, with this bit rate it is not viable to exchange large quantities of data. However, this is an adequate system to for example send commands to an AUV in mission.

The BPSK modulation was used so the system would be as precise as possible. However, the hardware of the system allowed for the use of Quadrature Phase Shift Keying (QPSK), if this digital modulation where to be used, it would be possible to increase the number of pseudo-random binary sequences used without increasing the length of the signal. And by extent, it would be possible to increase the transferred data.

## 6.1 Future Work

The following functionalities can be added to the developed system:

- Use more than one transmitter where each transmitter has a know position and sends a different signal. The receiver can then establish its distance to each one of the transmitters, and by knowing the position of each one, can then calculate its own position;

- Use a board to drive the transducer that can drive signals modulated in amplitude as well as phase, this way it is possible to use digital constellation such as 16 QAM. By doing this, it is possible to use a larger set of pseudo-random binary sequences without generating signals with a longer duration;
- Test the system in open sea at longer distances between transmitter and receiver to test the limits of the system.

# References


J. Aparicio, A. Jiménez, F. J. Álvarez, J. Ureña, C. D. Marziani, D. d. Diego, N. Cruz, and H. Campos. Accurate detection of spread-spectrum modulated signals in reverberant underwater environments. *Applied Acoustics*, 88:57 – 65, 2015. ISSN 0003-682X. doi: https://doi.org/10.1016/j.apacoust.2014.08.002.

J. Borden and J. DeArruda. Long range acoustic underwater communication with a compact auv. In *2012 Oceans*, pages 1–5, Oct 2012. doi: 10.1109/OCEANS.2012.6405091.

H. M. P. Cabrala. *Acoustic Modem for Underwater Communication*. Thesis, 2014. URL https://paginas.fe.up.pt/~ee09142/public/files/thesis.pdf.

Jingdong Chen, Jacob Benesty, and Yiteng(Arden) Huang. Time delay estimation in room acoustic environments: An overview. *EURASIP Journal on Advances in Signal Processing*, 2006(1): 026503, May 2006. ISSN 1687-6180. doi: 10.1155/ASP/2006/26503. URL https://doi.org/10.1155/ASP/2006/26503.

N. Cruz. Emissão de sinais acústicos subaquáticos, June 2007.

Daniel Dalskov and Søren Krarup Olesen. Locating acoustic sources with multilateration. *Master's*, 2014.

Safran Electronics Defense. Underwater navigation: reliable high-performance solutions | safran electronics & defense. https://www.safran-electronics-defense.com/naval-solutions/submarines/navigation-systems, 2018. (Accessed on 06/23/2018).

E. H. Dinan and B. Jabbari. Spreading codes for direct sequence cdma and wideband cdma cellular networks. *IEEE Communications Magazine*, 36(9):48–54, Sep 1998. ISSN 0163-6804. doi: 10.1109/35.714616.

H. Huang, Y. R. Zheng, and W. Duan. Pseudo-noise based time of arrival estimation for underwater acoustic sensor localization. In *OCEANS 2016 - Shanghai*, pages 1–5, April 2016. doi: 10.1109/OCEANSAP.2016.7485588.

M. W. Khan, Y. Zhou, and G. Xu. Modeling of acoustic propagation channel in underwater wireless sensor networks. In *The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014)*, pages 586–590, Nov 2014. doi: 10.1109/ICSAI.2014.7009354.

NOAA. What is ocean exploration and why is it important? http://oceanexplorer.noaa.gov/backmatter/whatisexploration.html, July 2014. (Accessed on 01/15/2018).

NOAA. How much of the ocean have we explored? https://oceanservice.noaa.gov/facts/exploration.html, October 2017. (Accessed on 01/15/2018).

M. Obara, G. Fischer, and Sangmok Lee. A monolithic time of arrival detector for acoustic signals. In *The 2002 45th Midwest Symposium on Circuits and Systems, 2002. MWSCAS-2002.*, volume 1, pages I–44–7 vol.1, Aug 2002. doi: 10.1109/MWSCAS.2002.1187149.

J. N. S. P. Oliveira. *Desenvolvimento De Um Sistema De Posicionamento Acústico Usbl E Validação Com Testes De Mar*. Thesis, 2009. URL https://fenix.tecnico.ulisboa.pt/downloadFile/395139419473/tese_final.pdf.

L. Paull, S. Saeedi, M. Seto, and H. Li. Auv navigation and localization: A review. *IEEE Journal of Oceanic Engineering*, 39(1):131–149, Jan 2014. ISSN 0364-9059. doi: 10.1109/JOE.2013.2278891.

M. d. F. Rosa. Autonomous surface vehicle based docking for an autonomous underwater vehicle. 2017. URL http://hdl.handle.net/10216/106731.

E. Rowan. Lbl underwater positioning. https://www.hydro-international.com/content/article/lbl-underwater-positioning, January 2008. (Accessed on 01/17/2018).

Daniele-Mining Surveyor, Mr Liggieri Sebastiano Giovanni-Surveyor, and Mr Spadaro. Underwater acoustic positioning system. 2013.

Hwee-Pink Tan, Roee Diamant, Winston K.G. Seah, and Marc Waldmeyer. A survey of techniques and challenges in underwater localization. *Ocean Engineering*, 38(14):1663 – 1676, 2011. ISSN 0029-8018. doi: https://doi.org/10.1016/j.oceaneng.2011.07.017. URL http://www.sciencedirect.com/science/article/pii/S0029801811001624.

J. F. R. Valente. *Real-Time Passive Acoustic Tracking of Underwater Vehicles*. Thesis, 2016. URL http://hdl.handle.net/10216/85105.

L. Zhou. *A Precise Underwater Acoustic Positioning Method Based on Phase Measurement*. Thesis, 2010. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.951.8601&rep=rep1&type=pdf.